普通高等教育
“十一五”国家级
规划教材

普通高等教育
电气工程与自动化类
“十一五”规划教材

COMPUTER SIMULATION AND CAD OF CONTROL SYSTEM

控制系统数字仿真与CAD

第3版

哈尔滨工业大学　张晓华　主编

机械工业出版社
CHINA MACHINE PRESS

TP273

普通高等教育“十一五”国家级规划教材
普通高等教育电气工程与自动化类“十一五”规划教材

★电路基础	东南大学	黄学良
电路实验教程	燕山大学	毕卫红
工程电磁场基础及应用	山东大学	刘淑琴
数字电子技术	中国计量学院	王秀敏
电子技术实验教程	天津大学	王萍
★计算机软件技术基础	哈尔滨工程大学	李金
通信技术基础（非通信类）	重庆邮电大学	鲜继清 刘焕淋 蒋青
★微型计算机原理及应用	西安交通大学	张彦斌
计算机网络与通信	清华大学	张曾科
★自动控制理论	合肥工业大学	王孝武 方敏 葛锁良
★自动控制理论	西安理工大学	刘丁
★现代控制理论基础（第2版）	合肥工业大学	王孝武
现代控制理论	浙江大学	赵光宙
自动控制原理（非自动化类）	浙江工业大学	王万良
信号分析与处理（第2版）	浙江大学	赵光宙
自动化概论	四川大学	赵曜
★电力电子技术（第5版）	西安交通大学	王兆安 刘进军
电力电子技术（少学时）	华南理工大学	张波
Power Electronics		吴斌
★电机及拖动基础（第4版）（上下册）	合肥工业大学	顾绳谷
电力拖动基础	四川大学	张代润
★电力拖动自动控制系统——运动控制系统（第4版）	上海大学	阮毅 陈伯时
电力传动控制系统	上海海事大学	汤天浩
★控制系统数字仿真与CAD（第3版）	哈尔滨工业大学	张晓华
★过程控制与自动化仪表（第2版）	西安理工大学	潘永湘
过程控制与自动化仪表	浙江大学	张宏建
过程控制系统	华东理工大学	俞金寿 孙自强
传感器与检测技术	清华大学	赵勇
计算机控制技术	沈阳大学	范立南
现场总线技术及应用	哈尔滨工业大学	佟为明
电磁兼容原理及应用	华中科技大学	熊蕊
★电气绝缘技术基础（第4版）	西安交通大学	曹晓珑
★电机学	重庆大学	韩力
电力工程	河海大学	鞠平
★供电技术（第4版）	西安理工大学	余健明 同向前 苏文成
智能控制理论及应用	湖南大学	王耀南 孙炜
智能电器	大连理工大学	邹积岩
建筑智能化系统	东北大学	吴成东
控制电机	山东大学	李光友 王建民 孙雨萍
机器人引论	清华大学	张涛
智能机器人引论	中国科学技术大学	关胜晓
嵌入式系统原理与应用	青岛大学	范延滨
数字图像处理与应用基础	西安理工大学	朱虹
电网络理论（图论 方程 综合）	浙江大学	周庭阳 张红岩
现代电机控制技术	沈阳工业大学	王成元 夏加宽 孙宜标
非线性电路理论	北京信息科技大学	刘小河

1. 本套教材陆续配套免费电子课件，欢迎选用本套教材的老师登录 www.cmpedu.com 注册下载或发邮件至 wbj@cmpbook.com 索取
2. 书名前标“★”号的为“普通高等教育‘十一五’国家级规划教材”

地址：北京市百万庄大街22号 邮政编码：100037
电话服务
社服务中心：(010)88361066 网络服务
销售一部：(010)68326294 门户网：http://www.cmpbook.com
销售二部：(010)88379649 教材网：http://www.cmpedu.com
读者服务部：(010)68993821 封面无防伪标均为盗版

ISBN978-7-111-28802-2
封面设计：王洪流
定 价：28.00 元



普通高等教育“十一五”国家级规划教材
普通高等教育电气工程与自动化类“十一五”规划教材

控制系统数字仿真与 CAD

第 3 版

主　编　张晓华
参　编　王华民
主　审　薛定宇



机械工业出版社

本书在概述了仿真技术发展进程的基础上，以 MATLAB 语言为平台，系统地阐述了数字仿真技术的基本概念、原理与方法，概括了控制系统分析与设计中的基本内容。全书共五章，主要包括：控制系统的实验方法、系统的数学描述与模型的建立、数字仿真的实现方法、控制系统 CAD 及其综合应用等内容。

书中涉及运动控制、过程控制、电磁与电力电子控制等系统的建模、分析与设计问题，其内容深入浅出，可读性较强；各章设有“问题与探究”一节，配有练习型、分析/设计型和探究型的习题，有助于激发学生的兴趣，拓展技术领域，以使读者进一步领会与掌握相关领域的内容。

本书系高等院校自动化专业本科生用教材，也可作为电气工程及其自动化、机械设计制造及其自动化等专业“仿真技术”类课程的教学用书。

图书在版编目（CIP）数据

控制系统数字仿真与 CAD/张晓华主编．—3 版．—北京：机械工业出版社，2009.12

普通高等教育“十一五”国家级规划教材

普通高等教育电气工程与自动化类“十一五”规划教材

ISBN 978-7-111-28802-2

Ⅰ．控… Ⅱ．张… Ⅲ．①控制系统－数字仿真－高等学校－教材②控制系统－计算机辅助设计－高等学校－教材 Ⅳ．TP273

中国版本图书馆 CIP 数据核字（2009）第 239037 号

机械工业出版社（北京市百万庄大街 22 号 邮政编码 100037）

策划编辑：于苏华 责任编辑：于苏华 版式设计：霍永明

封面设计：王洪流 责任校对：李秋荣 责任印制：乔 宇

北京机工印刷厂印刷

2010 年 2 月第 3 版第 1 次印刷

184mm×260mm · 15.75 印张 · 388 千字

标准书号：ISBN 978-7-111-28802-2

定价：28.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

电话服务

社服务中心：（010）88361066

销 售 一 部：（010）68326294

销 售 二 部：（010）88379649

读者服务部：（010）68993821

网络服务

门户网：http：//www.cmpbook.com

教材网：http：//www.cmpedu.com

封面无防伪标均为盗版

第 3 版前言

一、关于本书

1997 年 5 月，针对“数字仿真与 CAD 技术”的广泛应用与发展趋势，为满足本科生教学工作的需要，全国高等学校工业自动化专业教学指导委员会决定组织编写“控制系统数字仿真与 CAD”课程的本科生教材，并将其列为“九五”规划教材。1999 年，《控制系统数字仿真与 CAD》（第 1 版）由机械工业出版社出版，并于 2005 年修订。

《控制系统数字仿真与 CAD》出版 10 年以来，作为国内较早地将 MATLAB 语言融入仿真课程中的本科生教材，广泛地被读者评价为注重基础、内容精炼、可读性强、工程案例内容丰富、多媒体课件资料齐全，先后被国内 100 余所院校选作为仿真类课程的教材。

随着我国高等教育事业的发展，目前已由原来的多个本科专业合并而成自动化、电气工程与自动化、电气工程及其自动化、机械设计制造及其自动化等本科专业。同时，高等院校本科生的培养目标已向着厚基础、宽口径、重能力的方向发展，仿真技术类课程的内容也逐步从原理与软件学习拓展到工程案例教学方向上来。

因此，作为涉及自动控制原理、数值分析、控制系统设计等内容的仿真技术类课程，需要不断地与时俱进。

二、仿真技术类课程教学所面临的问题

1. 课程设置的必要性

仿真技术在其近 50 年的发展历史中，推动了几乎所有设计领域的革命，被喻为 20 世纪下半叶十大工程技术成就之一；如今，仿真技术已成为现代工程师应该掌握的基本技能之一。

作为联系自动控制理论、自动控制系统/设计、课程设计、毕业设计等教学环节的仿真技术类课程，其不仅可以使学生加强课程的学习效果，而且还可为学生在毕业设计中提供一个强有力的工具，有效加强教学中的实践性教学环节，提高学生的独立工作能力和创造性思维能力。

2. 新技术的不断涌现

作为以计算机系统、数值技术以及应用软件为基本手段的仿真技术，其内涵与其应用将会随着 IT 业的发展、各行业的需求和新技术的涌现而不断地向前发展。如何引领前沿技术进入仿真技术类课程的教学中来，以提高学生的学习兴趣，激发学生的创造意识，开阔他们的思维视野，是本课程面临的重要问题之一。

3. 能力的培养与训练

作为一类建立在若干先修课程与知识基础之上的应用型专业课，仿真技术类课程应该传授给学生些什么？笔者认为：把仿真技术这一“利器”传授给学生，使学生掌握其中的基本概念、基本原理和基本方法是本门课程最低层次的要求。同时，如何充分利用仿真工具，使学生加深理解所学的课程与知识内容，培养学生独立地分析问题与解决问题的能力，激发学生的创造意识，训练学生的思维方法，是仿真技术类课程教学所面临的深层次问题。

三、关于本书的修订

本书在第2版的基础上，主要有如下内容变动：

1）增加了“系统建模”内容：通过增加“三相电压型 PWM 整流器系统控制”与“磁悬浮轴承运动控制”两个电磁与电力电子系统的控制问题，进一步丰富了系统建模部分的案例内容。

2）增加“问题与探究”一节：本着“不学不成、不问不知”（汉王充）的教育理念，各章均设有“问题与探究”一节，以拓展我们的视野，为读者提供一个自主学习的空间。

3）删除“MATLAB 与 Simulink 基础”一章：鉴于目前 MATLAB 语言已在高等院校师生中广泛普及，本书为控制篇幅、突出重点而不再赘述；如在教学上有需求，建议参考相关教材。

4）配有多媒体课件（容量达 700M）：为便于选用本书的院校与教师有效地组织教学工作，凡选用本书作为仿真技术类课程教材的院校，均可获赠逐年更新的多媒体课件与电子教案。课件内容包括：各章多媒体课件/电子教案 PPT、各章习题解答、教学参考 PPT/图片/影像、参考文献、教学文件（教学大纲、实验指导书、课程报告）等。

作为一门时代特色鲜明、向学生传授系统分析与设计的“利器”、培养学生综合运用所学知识、勤于思考、勇于探索的作风的课程，笔者认为：能力比知识更重要，新技术的不断涌现会使得知识变得陈旧，而能力永恒；过程比结果更重要，随着问题与条件的改变，结果将会不同，而通过提出问题、分析问题、解决问题、归纳总结这样一个过程的训练，将会有效地塑造学生们科学的思维方式与工作习惯，而这将会使他们终生受益。因此，本书的主要目的在于：

1）通过向读者传授仿真技术，使读者清楚：随着新技术的不断发展，将会不断地产生更有效、更实用的“仿真工具”，因此我们应该不断地学习，以使自己能够与时俱进；

2）为读者讲明“仿真技术”中所涉及的基本原理、基本概念与基本方法，因为这些是我们能够有效运用“仿真工具”的理论基础。

3）给读者提出一些生动有趣、启迪思想的工程实际问题，创造一个自由畅想、激发创造的空间，以使大家从中体会到：“仿真技术”是我们学习、科研以及生活中不可缺少的工具。

如果本书能够在仿真技术类课程的教学工作中，使得教师“开阔思路、得心应手、提高效果”，使得学生“加深理解、掌握精髓、塑造思维”的话，那将是我们莫大的欣慰。

本书共分五章，其中第一、二、四、五章由哈尔滨工业大学张晓华教授编写，第三章由西安理工大学王华民教授编写；全书由张晓华教授统稿，东北大学薛定宇教授主审。

四、关于本书的使用

本书按授课30学时编写。对于计算机仿真技术基础课程，可选用本书的前四章（其中“问题与探究”一节的内容，可视具体情况灵活掌握）；一般学时数为20学时+4学时实验。对于“控制系统数字仿真与 CAD”课程，可选用全书内容（其中“问题与探究”一节内容、第五章的内容，可视具体情况灵活掌握）；一般学时数为30学时+8学时实验。

“仿真技术”是一门实践性较强的专业课，一般均要求安排一定量的上机实验。对于4学时的实验，可安排“水箱液位控制”或“双闭环直流调速系统设计”的仿真实验，以培养学生应用 MATLAB 语言解决实际问题的能力；对于8学时的实验，可在“双闭环直流调

速系统设计”仿真实验基础上，安排“基于双闭环PID控制的一阶倒立摆控制系统设计”的仿真实验，重点在于培养学生的综合应用能力。

作为以提高学生能力为主要目的仿真技术类课程，其一般为考查课。因此，笔者建议：本门课程的考核以写作报告的形式为主，如“实验报告”或“课程报告”，重点考核学生解决实际问题的能力、归纳总结能力以及科技论文的撰写能力。

五、致谢

在本书的成稿与面世过程中，得到了以下同仁的热诚支持与帮助：

哈尔滨工业大学教学发展基金的资助；

哈尔滨工业大学电气工程及自动化学院；

西安理工大学自动化与信息工程学院；

东北大学薛定宇教授；

哈尔滨工业大学赵旖旎、齐海明、郭源博、姚文昊、林乐天研究生；

机械工业出版社。

在此，一并致以衷心的谢意。

由于编者水平有限，错误与不当之处在所难免，殷切期望广大读者批评指正。

信函请至：哈尔滨工业大学电气学院354信箱　张晓华　收，邮编150001；

E-mail：xh _ zhang@ hit. edu. cn。

编者

2009年9月

目　录

第 3 版前言

第一章　概述 …… 1

第一节　控制系统的实验方法 …… 1

一、解析法 …… 1

二、实验法 …… 1

三、仿真实验法 …… 2

第二节　仿真实验的分类与性能比较 …… 2

一、按模型分类 …… 3

二、按计算机类型分类 …… 4

第三节　系统、模型与数字仿真 …… 6

一、系统的组成与分类 …… 6

二、模型的建立及其重要性 …… 7

三、数字仿真的基本内容 …… 8

第四节　控制系统 CAD 与数字仿真软件 …… 9

一、CAD 技术的一般概念 …… 9

二、控制系统 CAD 的主要内容 …… 10

三、数字仿真软件 …… 10

第五节　仿真技术的应用与发展 …… 16

一、仿真技术在工程中的应用 …… 16

二、应用仿真技术的重要意义 …… 17

三、仿真技术的发展趋势 …… 18

第六节　问题与探究——虚拟现实与仿真技术 …… 18

一、虚拟现实技术 …… 18

二、虚拟现实仿真技术 …… 22

三、基于虚拟样机的球棒控制系统仿真 … 25

小结 …… 27

习题 …… 27

第二章　控制系统的数学描述 …… 30

第一节　控制系统的数学模型 …… 30

一、控制系统数学模型的表示形式 …… 30

二、数学模型的转换 …… 32

三、线性时不变系统的对象数据类型描述 … 34

四、控制系统建模的基本方法 …… 35

第二节　控制系统建模实例 …… 39

一、独轮自行车实物仿真问题 …… 39

二、龙门起重机运动控制问题 …… 42

三、水箱液位控制问题 …… 46

四、燃煤热水锅炉控制问题 …… 49

五、三相电压型 PWM 整流器系统控制问题 …… 51

六、磁悬浮轴承运动控制问题 …… 56

第三节　实现问题 …… 61

一、单变量系统的可控标准型实现 …… 61

二、控制系统的数字仿真实现 …… 63

第四节　常微分方程数值解法 …… 63

一、数值求解的基本概念 …… 64

二、数值积分法 …… 65

三、关于数值积分法的几点讨论 …… 69

第五节　数值算法中的“病态”问题 …… 72

一、“病态”常微分方程 …… 72

二、控制系统仿真中的“病态”问题 …… 74

三、“病态”系统的仿真方法 …… 75

第六节　数字仿真中的“代数环”问题 …… 76

一、问题的提出 …… 76

二、“代数环”产生的条件 …… 77

三、消除“代数环”的方法 …… 78

第七节　问题与探究——电力电子器件建模问题 …… 81

一、问题提出 …… 81

二、建模机理 …… 81

三、问题探究 …… 84

小结 …… 85

习题 …… 86

第三章　控制系统数字仿真的实现 …… 88

第一节　控制系统的结构及其拓扑描述 …… 88

一、控制系统常见的典型结构形式 …… 88

二、控制系统的典型环节描述 …… 89

三、控制系统的连接矩阵 …… 90

第二节　面向系统结构图的数字仿真 …… 92

一、典型闭环系统的数字仿真 …… 92

二、复杂连接的闭环系统数字仿真 …… 97

第三节　环节的离散化与非线性系统的数字仿真 …… 103

一、连续系统的离散化模型法 …… 103
二、非线性系统的数字仿真 …… 112
第四节 计算机控制系统的数字仿真 …… 119
一、采样控制系统的数学描述 …… 120
二、采样控制系统的仿真方法 …… 123
三、采样控制系统仿真程序实现 …… 126
第五节 问题与探究——一类非线性控制系统数字仿真的效率问题 …… 132
一、问题提出 …… 132
二、问题分析 …… 133
三、几点讨论 …… 139
小结 …… 139
习题 …… 140
第四章 控制系统 CAD …… 142
第一节 概述 …… 142
第二节 经典控制理论 CAD …… 143
一、控制系统固有特性分析 …… 143
二、控制系统的设计方法 …… 146
三、控制系统的优化设计 …… 154
第三节 基于双闭环 PID 控制的一阶倒立摆控制系统设计 …… 159
一、系统模型 …… 159
二、模型验证 …… 161
三、双闭环 PID 控制器设计 …… 165
四、仿真实验 …… 170
五、结论 …… 173
第四节 现代控制理论 CAD …… 173
一、线性二次型最优控制器设计 …… 174
二、模型参考自适应控制系统设计 …… 182
第五节 基于时间最优控制的起重机防摆控制技术研究 …… 188
一、问题的提出 …… 188
二、时间最优控制 …… 189
三、系统建模 …… 190
四、模型验证 …… 190
五、时间最优控制策略 …… 192
六、仿真实验 …… 194
七、结论 …… 196
第六节 问题与探究——“球车系统”的建模与控制问题 …… 197
一、问题提出 …… 197
二、系统建模 …… 197
三、问题探究 …… 198
小结 …… 198
习题 …… 199
第五章 数字仿真技术的综合应用 …… 202
第一节 直流电动机双闭环调速系统设计中的若干问题 …… 202
一、双闭环 V-M 调速系统的目的 …… 202
二、关于积分调节器的饱和非线性问题 … 203
三、关于 ASR 与 ACR 的工程设计问题 … 203
四、双闭环 V-M 调速系统的动态分析 … 205
第二节 数字 PID 调节器的鲁棒性设计方法 …… 207
一、数字 PID 调节器的鲁棒性设计 …… 207
二、“高精度齿轮量仪”位置伺服系统控制器设计 …… 209
第三节 “水箱系统”液位控制的仿真研究 …… 211
一、系统建模 …… 211
二、数字仿真 …… 211
三、结果分析 …… 211
第四节 一阶倒立摆系统的双闭环模糊控制方案 …… 212
一、引言 …… 212
二、模糊理论中的几个基本概念 …… 213
三、一阶倒立摆系统的双闭环模糊控制 …… 214
四、仿真实验 …… 216
五、结论 …… 223
第五节 基于单位功率因数的 PWM 整流器控制系统设计 …… 223
一、引言 …… 223
二、滑模变结构控制 …… 224
三、系统建模与模型验证 …… 225
四、基于滑模变结构控制的 PWM 整流器控制系统设计 …… 226
五、仿真实验 …… 230
六、结论 …… 234
第六节 问题与探究——两轮电动车自平衡控制问题 …… 234
一、问题提出 …… 234
二、系统建模 …… 236
三、问题探究 …… 237
小结 …… 238
习题 …… 238
参考文献 …… 241

第一章　概　　述

第一节　控制系统的实验方法

在工程设计与理论学习过程中，我们会接触到许多控制系统的分析、综合与设计问题，需要对相应的系统进行实验研究，概括起来有解析法、实验法与仿真实验法三种实验方法。

一、解析法

所谓解析法，就是运用已掌握的理论知识对控制系统进行理论上的分析、计算。它是一种纯理论意义上的实验分析方法，在对系统的认识过程中具有普遍意义。

例如，在研究汽车轮子悬挂系统的减振器性能及其弹簧参数变化对汽车运动性能的影响时，可从动力学角度分析，将系统等效为图 1-1 所示模型形式，进而得出描述该系统动态过程的二阶常微分方程

$$a\frac{d^2x}{dt^2} + b\frac{dx}{dt} + cx = F(t) \qquad (1\text{-}1)$$

对于式（1-1）的分析求解显然就是一个纯数学解析问题。但是，在许多工程实际问题中，由于受到理论的不完善性以及对事物认识的不全面性等因素的影响（例如“黑箱”问题、“灰箱”问题等），所以解析法往往有很大的局限性。

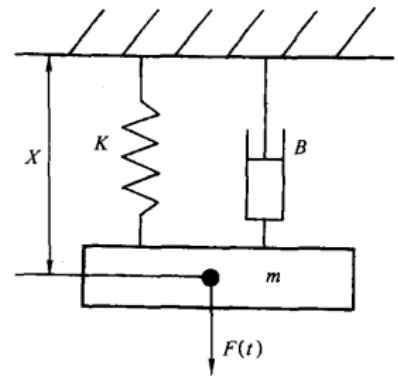


图 1-1　悬挂系统动力学模型

二、实验法

对于已经建立的（或已存在的）实际系统，利用各种仪器仪表与装置，对系统施加一定类型的信号（或利用系统中正常的工作信号），通过测取系统响应来确定系统性能的方法称之为实验法。它具有简明、直观与真实的特点，在一般的系统分析与测试中经常采用。

图 1-2 给出的是一带传动试验机转速控制系统，其动态性能 $n(t)$ 及静态性能 $n(I_d)$ 均可通过实验的方法测得，图 1-3 是其静特性的测量结果。

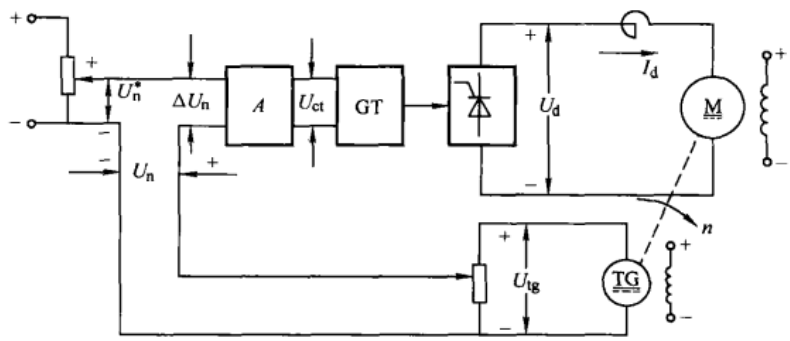


图 1-2　带传动试验机转速控制系统

但是，由于种种原因，这种实验方法在实际中常常难以实现。归纳起来有如下几方面的原因：

1）对于控制系统的设计问题，由于实际系统还没有真正的建立起来，所以不可能在实际的系统上进行实验研究。

2）实际系统上不允许进行实验研究。比如在化工控制系统中，随意改变系统运行的参数，往往会导致最终成品的报废，造成巨额损失，类似的问题还有许多。

3）费用过高、具有危险性、周期较长。比如：大型加热炉、飞行器及原子能利用等问题的实验研究。

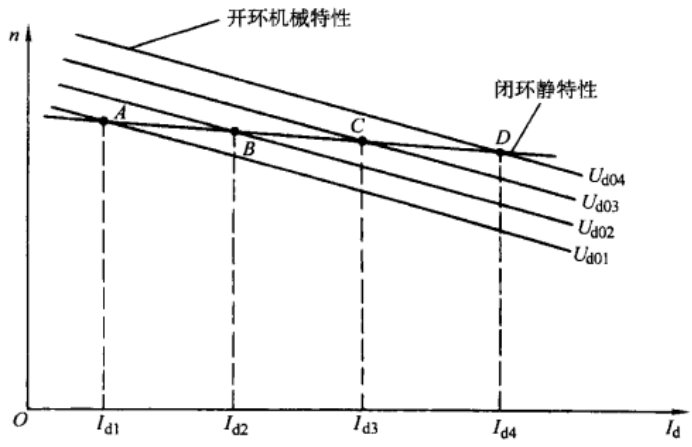


图 1-3 转速控制系统静特性

鉴于上述原因，在模型上进行的仿真实验研究方法逐渐成为对控制系统进行分析、设计与研究的十分有效的方法。

三、仿真实验法

仿真实验法就是在模型上（物理的或数学的）所进行的系统性能分析与研究的实验方法，它所遵循的基本原则是相似原理。

系统模型可分为两类，一类为物理模型，另一类是数学模型。例如，在飞行器的研制中，将其放置在“风洞”之中进行的实验研究，就是模拟空中情况的物理模型的仿真实验研究，其满足“环境相似”的基本原则。又如，在船舶设计制造中，常常按一定的比例尺缩小建造一个船舶模型，然后将其放置在水池中进行各种动态性能的实验研究，其满足“几何相似”的基本原则，是模拟水中情况的物理模型的仿真实验研究。

在物理模型上所作的仿真实验研究具有效果逼真、精度高等优点；但是，其或者造价高昂，或者耗时过长，不宜为广大的研究人员所接受，大多是在一些特殊场合下（比如，导弹或卫星一类飞行器的动态仿真，发电站综合调度仿真与培训系统等）采用。

随着计算机与微电子技术的飞速发展，人们越来越多地采用数学模型在计算机（数字的或模拟的）上进行仿真实验研究。在数学模型上所进行的仿真实验是建立在“性能相似”的基本原则之上的。因此，通过适当的手段与方法建立高精度的数学模型是其前提条件。

第二节 仿真实验的分类与性能比较

由于仿真实验是利用模型（物理的或数学的）来进行系统动态性能研究的实验，其中绝大多数都要应用计算机（模拟的或数字的），因此其分类方式以及相应的称呼均有所不同。下面仅就常用的几种情况进行说明。

一、按模型分类

当仿真实验所采用的模型是物理模型时，称之为物理仿真；是数学模型时，称之为数学仿真。

事实上，人们经常根据仿真实验中有无实物介入以及与时间的对应关系将模型分类进一步地细化，其可归纳成图 1-4 所示的情况。由图可见，物理仿真总是有实物介入的，具有实时性与在线的特点。因此，仿真系统具有构成复杂、造价较高等特点，图 1-5 给出了某卫星姿态控制的实物仿真系统原理，从中可略见一斑。数学仿真是在计算机上进行的，具有非实时性与离线的特点，是一种经济、快捷与实用的实验方法。

本书重点讨论基于数学模型的数字仿真问题。

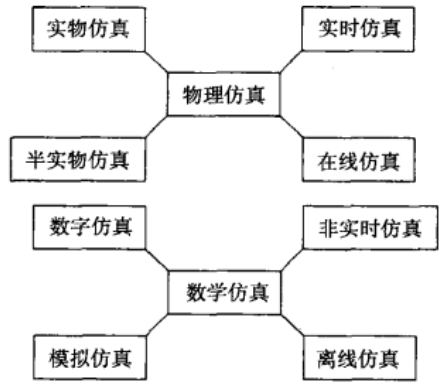


图 1-4 按模型分类的几种情况

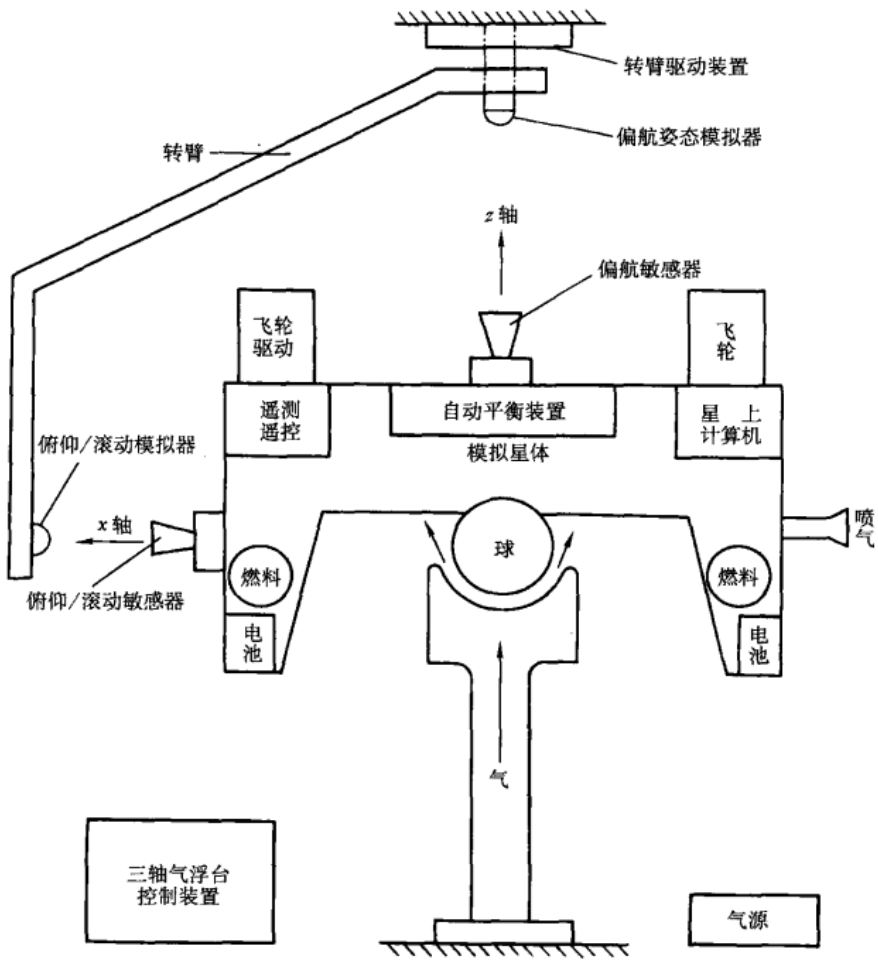


图 1-5 卫星姿态控制的实物仿真系统原理图

二、按计算机类型分类

由于数学仿真是在计算机上进行的，所以视计算机的类型以及仿真系统的组成不同可有多种仿真形式。

（一）模拟仿真

采用数学模型在模拟计算机上进行的实验研究称之为模拟仿真。模拟计算机的组成如图 1-6 所示，其中“运算部分”是它的核心，它是由我们熟知的“模拟运算放大器”为主要部件所构成的，能够进行各种线性与非线性函数运算的模拟单元。下面的例子说明了模拟仿真实验的实现过程。

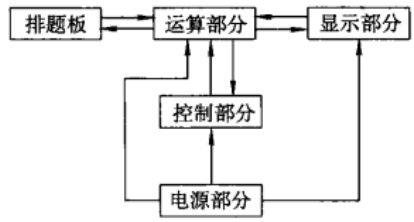


图 1-6 模拟计算机的组成

例 1-1 在图 1-1 所示系统中，若初始条件为 $\dot{X}(t)|_{t=0}=\dot{X}(o)=\alpha$，$X(t)|_{t=0}=X(o)=\beta$，试分析参数 B 对系统振动特性的影响。

解 对于式(1-1)，不难确定：$a=m$，$b=B$，$c=K$，则有

$$\ddot{X}(t)=-\frac{B}{m}\dot{X}(t)-\frac{K}{m}X(t)+\frac{1}{m}F(t) \tag{1-2}$$

据式（1-2）可有图 1-7 所示的模拟仿真结构图，依据它在模拟计算机排题板上即可进行排版及做仿真实验。

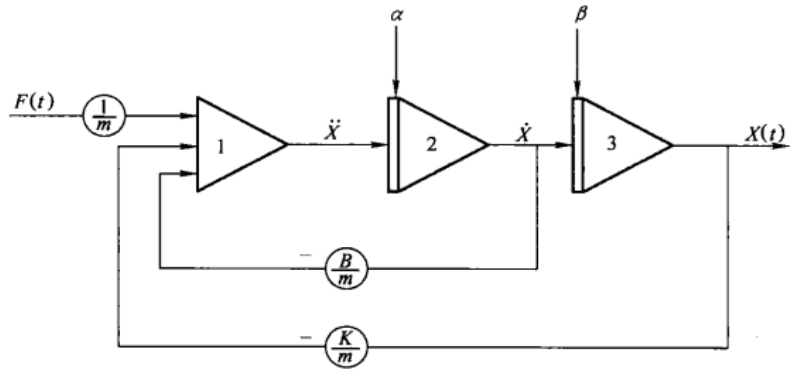


图 1-7 模拟仿真结构图

若 $F(t)=1(t)$，则当参数 B 取不同值时，有图 1-8 所示仿真结果。从中可见，适当选择 B 值可以使系统减小或消除振动，提高乘坐汽车的舒适性。这一结果与解析法分析结果是一致的。阻尼系数 B 值过小时系统易产生振动。

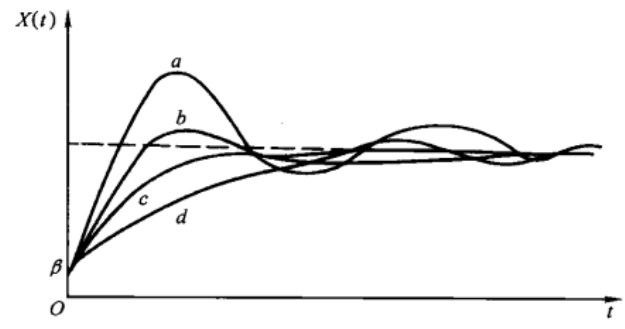


图 1-8 动态仿真结果

模拟仿真具有如下优缺点：

1）描述连续的物理系统的动态过程比较自然而逼真。

2）仿真速度极快，失真小，结果可信度高。

3）受元器件性能的影响，仿真精度较低。

4）对计算机控制系统（采样控制系统）的仿真较困难。

5）仿真实验过程的自动化程度较低。

（二）数字仿真

采用数学模型，在数字计算机上借助于数值计算的方法所进行的仿真实验称之为数字仿真。数字仿真具有简便、快捷、成本低的特点，同时还具有如下优缺点：

1）计算与仿真的精度较高。由于计算机的字长可以根据精度要求来“随意”设计，因此从理论上讲系统数字仿真的精度可以是无限的。但是，由于受到误差积累、仿真时间等因素的影响，其精度不宜定得过高。

2）对计算机控制系统的仿真比较方便。

3）仿真实验的自动化程度较高，可方便地实现显示、打印等功能。

4）计算速度比较低，在一定程度上影响到仿真结果的可信度。因此，其对一些“频响”较高的控制系统进行仿真时具有一定的困难。

随着计算机技术的发展，“速度问题”会在不同程度上予以改进与提高，因此可以说数字仿真技术有着极强的生命力。

（三）混合仿真

通过上面的介绍可以看到，模拟仿真与数字仿真各有优缺点，同时其优缺点可以互补，由此就产生了将这两种方法结合起来的混合仿真实验系统，简称混合仿真，其主要应用于下述情况：

1）要求对控制系统进行反复迭代计算时。例如：参数寻优、统计分析等。

2）要求与实物连接进行实时仿真，同时又有一些复杂函数的计算问题。

3）对于一些计算机控制系统的仿真问题。此时，数字计算机用于模拟系统中的控制器，而模拟计算机用于模拟被控对象。

混合仿真集中了模拟仿真与数字仿真的优点，其缺点是系统构成复杂、造价偏高。

（四）全数字仿真

对于计算机控制系统的仿真问题，在实际应用中为简化系统构成，对象的模拟也可用一台数字计算机来实现，用软件来实现对象各种机理的模拟，如图 1-9 所示。从中可见，控制计算机系统是真实系统，即今后要实际应用之；而仿真计算机是用来模拟被控对象的，可用软件灵活构成各种线性及非线性特性，因此全数字仿真系统具有灵活、多变、构成简便的特点。

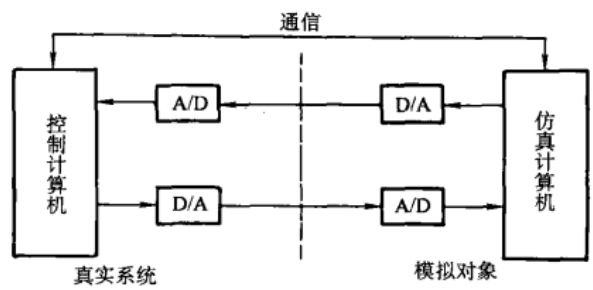


图 1-9 全数字仿真系统原理图

在全数字仿真中，若想进一步降低仿真系统成本，或仅用其作理论研究，则图 1-9 中的 A/D 与 D/A 接口电路部分可以去掉，用网络通信的方法实现控制器与模拟对象之间的信息交换，其在复杂系统数字仿真加速方法上具有独到之处。

（五）分布式数字仿真

对于算法复杂的大型数字仿真问题，单一的或仅用两台 PC 进行数字仿真往往受到速度与精度这一对矛盾因素的影响，尽管数字计算机单机的运行速度在不断提升，这一矛盾始终困扰数字仿真技术的推广及深入的应用。大型（或巨型）计算机虽然具有卓越的性能，但其价格限制了其市场范围。

那么如何用普通 PC 来解决数字仿真中的加速与精度的提高问题呢？现代计算机网络技术为其开辟了新途径。图 1-10 给出了基于网络技术实现的分布式数字仿真系统。从中可见，数字仿真系统将所研究的问题分布成若干个子系统，分别在主站与各分站的计算机上同时运行，其有用数据通过网络与主站进行信息交换，在网络通信速度足够高的条件下，分布式数字仿真系统具有近似的多 CPU 并行计算机的性能，使仿真速度与精度均可有所保证，而成本却相对低很多，这是一种简便有效地解决复杂系统数字仿真问题的方法。

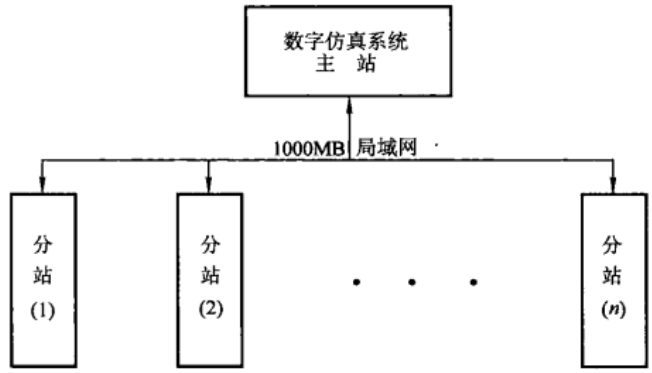


图 1-10 分布式数字仿真系统

第三节 系统、模型与数字仿真

在进行数字仿真实验时，对实际系统的认识，对系统模型的理解以及在计算机上的实现是一个有机的整体，每个环节都不同程度地对最终结果有所影响。因此，我们有必要对它们深入了解与掌握。

一、系统的组成与分类

所谓系统就是由一些具有特定功能的、相互间以一定规律联系着的物体（又称子系统）所构成的有机整体。

（一）组成系统的三个要素——实体、属性和活动

（1）实体 就是存在于系统中的具有确定意义的物体。比如电力拖动系统中的执行电动机、热力系统中的控制阀等。

（2）属性 即实体所具有的任何有效特征。比如温度、控制阀的开度及传动系统的速度等。

（3）活动 系统内部发生的任何变化过程称之为内部活动；而系统外部发生的对系统产生影响的任何变化过程称之为外部活动。比如：控制阀的开启为热力系统的内部活动，电网电压的波动为电力拖动系统的外部活动（即外部扰动）。

（二）系统具有的三种特性——整体性、相关性和隶属性

（1）整体性 即系统中的各部分（子系统）不能随意分割。比如任何一个闭环控制系统的组成中，对象、传感器及控制器缺一不可。因此，系统的整体性是一个重要特性，直接影响系统功能与作用。

（2）相关性 即系统中的各部分（子系统）以一定的规律和方式相联系，由此决定了其特有的性能。比如电动机调速系统是由电动机、测速机、PI 调节器及功率放大器等组成，并形成了电动机能够调速的特定性能。

（3）隶属性 一般情况下，有些系统并不像控制系统（由人工制成的）那样可清楚地分出系统的“内部”与“外部”，它们常常需要根据所研究的问题来确定哪些属于系统的内部因素，哪些属于系统的外界环境，其界限也常常随不同的研究目的而变化，将这一特性称之为隶属性。分清系统的隶属界限是十分重要的，它往往可使系统仿真问题得以简化，有效地提高仿真工作的效率。

（三）系统的分类

系统的分类可有多种形式，下面是以“时间”作为依据的分类情况：

系统 { 连续系统；离散系统 { 离散时间系统；离散事件系统 }；混合系统 }

（1）连续系统 系统中的状态变量随时间连续变化的系统为连续系统。如电动机速度控制系统、锅炉温度调节系统等。

（2）离散时间系统 系统中状态变量的变化仅发生在一组离散时刻上的系统为离散时间系统。如计算机系统。

（3）离散事件系统 系统中状态变量的改变是由离散时刻上所发生的事件所驱动的系统为离散事件系统。如大型仓储系统中的“库存”问题，其“库存量”是受“入库”、“出库”事件的随机变化的影响的。

离散事件系统的仿真问题本书未涉及，有兴趣的读者可参阅有关文献。

（4）连续离散混合系统 若系统中一部分是连续系统，而另一部分是离散系统，其间有连接环节将两者联系起来，则称之为连续离散混合系统。如计算机控制系统，通常情况下其对象为连续系统，而控制器为离散时间系统。

本书中所述的“离散系统”均指离散时间系统。

二、模型的建立及其重要性

（一）模型

系统模型是对系统的特征与变化规律的一种定量抽象，是人们用以认识事物的一种手段（或工具），一般有以下几种：

模型 { 物理模型；数学模型；描述模型 }

对于物理模型与数学模型，我们已有所了解，下面着重谈一下“描述模型”。

所谓描述模型是一种抽象的（无实体的），不能或很难用数学方法描述的，而只能用语言（自然语言或程序语言）描述的系统模型。

随着科学技术的发展，在许多系统中都存在着“精确”与“实现”之间的矛盾问题，即若过分追求模型的精确（即严格的数学模型），则实际中往往很难实现。因此，为了有效

地对一类复杂系统实现控制，人们已不再单纯地追求“数学模型”，而是建立起基于“经验”或“知识”的描述模型。例如，在模糊（Fuzzy）控制系统中，人们对控制对象的描述就是一组基于“经验”的 If-then-else 语句的描述。

描述模型是系统模型由“粗”向“精”转换过程中的一个中间模型，随着人们对系统行为的不断深入认识，其最终将被精确的数学模型所取代。

（二）模型的建立

建立系统模型就是（以一定的理论为依据）把系统的行为概括为数学的函数关系。其包括以下内容：

1）确定模型的结构，建立系统的约束条件，确定系统的实体、属性与活动。

2）测取有关的模型数据。

3）运用适当理论建立系统的数学描述，即数学模型。

4）检验所建立的数学模型的准确性。

（三）系统建模的重要性

由于控制系统的数字仿真是以其“数学模型”为前提的，所以对于仿真结果的“可靠性”来讲，系统建模至关重要，它在很大程度上决定了数字仿真实验的“成败”。

长期以来，由于人们对系统建模重视不够，使得数字仿真技术的应用仅仅限于“理论上的探讨”，缺乏对实际工作的指导与帮助，因而在一部分人的思想概念中产生了“仿真结果不可信”或“仿真用处不大”的错误认识。

现代的数字仿真技术已日趋完善地向人们提供强有力的仿真软件工具，从而对“系统建模”的要求越来越高，因此应予以充分的重视与熟练的掌握。

三、数字仿真的基本内容

通常情况下，数字仿真实验包括三个基本要素，即实际系统、数学模型与计算机。联系这三个要素则有三个基本活动，即模型建立、仿真实验与结果分析。以上所述三要素及三个基本活动的关系可用图 1-11 来表示。由图可见，将实际系统抽象为数学模型，称之为一次模型化，它还涉及系统辨识技术问题，统称为建模问题；将数学模型转换为可在计算机上运行的仿真模型，称之为二次模型化，这涉及到仿真技术问题，统称为仿真实验。

长期以来，仿真领域的研究重点一直放在仿真模型的建立这一活动上（即二次模型化问题），并因此产生了各种仿真算法及工具软件，而对于模型建立与仿真结果的分析问题重视不够，因此使得当一个问题提出后，需要较长的时间用于建模。同时，仿真结果的分析常常需要一定的经验，这对于进行仿真实验的工程技术人员来讲是有困难的，其结果造成仿真结果不真实、可信度低等问题，这些问题有碍于数字仿真技术的推广应用。

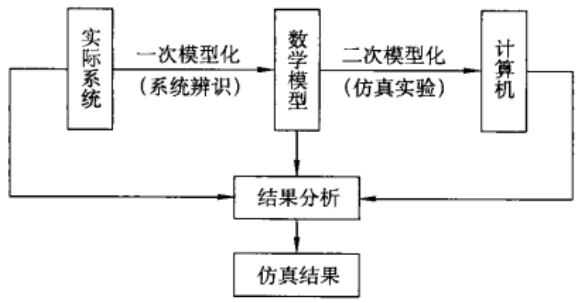


图 1-11 数字仿真的基本内容

综上所述，仿真实验是建立在模型这一基础之上的，对于数字仿真要完善建模、仿真实验及结果分析体系，以使仿真技术成为控制系统分析、设计与研究的有效工具。

第四节 控制系统 CAD 与数字仿真软件

计算机辅助设计（Computer Aided Design）技术，即 CAD 技术是随着计算机技术的发展应运而生的一门应用型技术，至今已有近 40 年的历史。1989 年，美国评出了科技领域近 25 年间最杰出的十项工程技术成就，将 CAD/CAM 技术列为第四项，称之为“推动了几乎所有设计领域的革命”。

孟子曰：“工欲善其事，必先利其器”，CAD 技术已成为当今推动技术进步与产品更新换代不可缺少的有力工具。

一、CAD 技术的一般概念

（一）什么是 CAD 技术

CAD 技术就是将计算机高速而精确的计算能力、大容量存储和处理数据的能力与设计者的综合分析、逻辑判断以及创造性思维结合起来，用以加快设计进程、缩短设计周期、提高设计质量的技术。

CAD 不是简单地使用计算机代替人工计算、制图等“传统的设计方法”，而是通过 CAD 系统与设计者之间强有力的“信息交互”作用，从本质上增强设计人员的想象力与创造力，从而有效地提高设计者的能力与设计结果的水平。在近 20 年的发展历史中，汽车制造业的推陈出新、服装加工业的层出不穷以及航空航天领域的卓越成就等，无不与 CAD 技术的发展有着密切的联系。

因此，CAD 技术中所涉及的“设计”应该是以提高社会生产力水平、加快社会进步为目的的创造性的劳动。

（二）CAD 系统的组成

CAD 系统通常是由应用软件、计算机、外围设备以及设计者本身（即用户）组成的，它们之间的关系如图 1-12 所示。其中，应用软件是 CAD 系统的“核心”内容，在不同的设计领域有相应的 CAD 应用软件，例如，机械设计中有 Auto CAD 软件，控制系统设计中有 MATLAB 软件（及相应工具箱）；计算机是 CAD 技术的“基础”，随着单机性能的不断提高，CAD 技术将更广泛地为各行业所采用；外围设备是人-机信息交换的手段。显示技术与绘图打印技术的不断发展为 CAD 技术提供了丰富多彩的表现形式，在提高设计者的想象力、创造力以及最终结果的展现等方面具有重要意义。

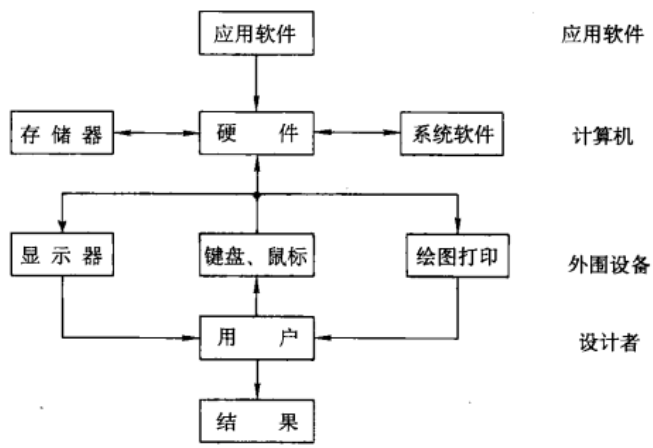


图 1-12 CAD 系统的组成

（三）怎样面对 CAD 技术

由于 CAD 技术涉及数字仿真、计算方法、显示与绘图以及计算机等诸多内容，作为

CAD 技术的使用者，我们应注意以下几方面的问题：

1）注重对所涉及内容基本概念的理解与掌握，它是我们今后能够进行创造性思维与逻辑推理的理论基础。

2）选择数值可靠、性能优越的应用软件作为 CAD 系统的“核心”，以使设计结果具有实际意义。

3）将理论清晰、概念明确但分析计算复杂的工作交给计算机来做，作为设计者应主要从事具有创造性的设计工作。

控制系统 CAD 作为 CAD 技术在自动控制理论及自动控制系统分析与设计方面的应用分支，是本门课程的另一个重要内容。

二、控制系统 CAD 的主要内容

CAD 技术为控制系统的分析与设计开辟了广阔天地，它使得原来被人们认为难以应用的设计方法成为可能。一般认为，控制系统分析与设计方法有两类，即频域法（又称变换法）和时域法（又称状态空间法）。

（一）频域法

频域法属经典控制理论范畴，主要适用于单输入单输出系统。频域法借助于传递函数、劳斯判据、Bode 图、Nyquist 图及根轨迹等概念与方法来分析系统动态特性和稳态性能，设计系统校正装置的结构，确定最优的装置参数。

（二）时域法

时域法为现代控制理论内容，适用于多变量系统的分析与设计。其主要内容有：①线性二次型最优控制规律与卡尔曼滤波器的设计；②闭环系统的极点配置；③状态反馈与状态观测器的设计；④系统稳定性、能控性、能观性及灵敏度分析等。

此外，自适应控制、自校正控制以及最优控制等现代控制策略都可利用 CAD 技术实现有效的分析与设计。

三、数字仿真软件

作为控制系统 CAD 技术中的“核心”内容——应用软件，数字仿真软件始终为该领域研究开发的热点，人们总是以最大限度地满足使用者（特别是工程技术人员）方便、快捷、精确的需求为目的，不断地使数字仿真软件推陈出新。

（一）数字仿真软件的发展

随着计算机与数字仿真技术的发展，数字仿真软件经历了以下四个阶段：

（1）程序编制阶段　即在人们利用数字计算机进行仿真实验的初级阶段时，所有问题（如微分方程求解、矩阵运算、绘图等）都是仿真实验者用高级算法语言（如 BASIC、FORTRAN、C 等）来编写。往往是几百条语句的编制仅仅解决了一个“矩阵求逆”一类的基础问题，人们大量的精力不是放在研究“系统问题”如何，而是过多地研究软件如何编制、其数值稳定性如何等旁支问题，其结果使得仿真工作的效率较低，数字仿真技术难以为众人所广泛应用。

（2）程序软件包阶段　针对“程序编制阶段”所存在的问题，许多系统仿真技术的研究人员将他们编制的数值计算与分析程序以“子程序”的形式集中起来形成了“应用子程

序库”，又称为“应用软件包”（以便仿真实验者在程序编制时调用）。这一阶段中的许多成果为数字仿真技术的应用奠定了基础，但还是存在着使用不便、调用繁琐、专业性要求过强、可信度低等问题，人们已开始认识到，建立具有专业化与规格化的高效率的“仿真语言”是十分必要的，以使数字仿真技术真正成为一种实用化的工具。

（3）交互式语言阶段 从人-机之间信息交换便利的角度出发，将数字仿真所涉及的问题上升到“语言”的高度所进行的软件集成，其结果就产生了交互式的“仿真语言”。仿真语言与普通高级算法语言（如 C、FORTRAN 等）的关系就如同 C 语言与汇编语言的关系一样，人们在用 C 语言在进行乘（或除）法运算时不必去深入考虑乘法是如何实现的（已有专业人员周密处理之）；同样，仿真语言可用一条指令实现“系统特征值的求取”，而不必考虑是用什么算法以及如何实现等低级问题。

当今具有代表性的仿真语言有：瑞典 Lund 工学院的 SIMNON 仿真语言、IBM 公司的 CSMP 仿真语言以及 ACSL、TSIM、ESL 等。20 世纪 80 年代初，由美国学者 Cleve Moler 等人推出的交互式 MATLAB 语言以它独特的构思与卓越的性能为控制理论界所重视，现已成为控制系统 CAD 领域最为普及与流行的应用软件。

（4）模型化图形组态阶段 尽管仿真语言将人-机界面提高到“语言”的高度，但是对于从事控制系统设计的专业技术人员来讲还是有许多不便，他们似乎对基于模型的图形化（如框图）描述方法更亲切。随着“视窗”（Windows）软件环境的普及，基于模型化图形组态的控制系统数字仿真软件应运而生，它使控制系统 CAD 进入到一个崭新的阶段。目前，最具代表性的模型化图形组态软件当数美国 Math Works 软件公司 1992 年推出的 Simulink，它与该公司著名的 MATLAB 软件集成在一起，成为当今最具影响力的控制系统 CAD 软件。

（二）MATLAB

MATLAB 是美国 Math Works 公司的软件产品。

20 世纪 80 年代初期，美国的 Cleve Moler 博士（数值分析与数值线性代数领域著名学者）在教学与研究工作中充分认识到当时的科学分析与数值计算软件编制工作的困难所在，便构思开发了名为 MATrix LABoratory（矩阵实验）的集命令翻译、科学计算于一体的交互式软件系统，其有效地提高了科学计算软件编制工作的效率，迅速成为人们广泛应用的软件工具。MATLAB 作为原名的缩写成为后来由 Moler 博士及一批优秀数学家与软件专家组成的 Math Works 公司软件产品的品牌。

尽管 MATLAB 一开始并不是为控制系统的设计者们设计的，但是其一出现便以它“语言”化的数值计算、较强的绘图功能、灵活的可扩充性和产业化的开发思路很快就为自动控制界研究人员所瞩目。目前，在自动控制、图像处理、语言处理、信号分析、振动理论、优化设计、时序分析与统计学、系统建模等领域，由著名专家与学者以 MATLAB 为基础开发的实用工具箱极大地丰富了 MATLAB 的内容，使之成为国际上最为流行的软件品牌之一。

应该指出的是，尽管 MATLAB 在功能上已经完全具备了计算机语言的结构与性能，人们将其简称为：“MATLAB 语言”，但是由于其编写出来的程序并不能脱离 MATLAB 环境而独立运行，所以严格地讲，MATLAB 并不是一种计算机语言，而是一种高级的科学分析与计算软件。

（三）Simulink

Simulink 是美国 Math Works 软件公司为其 MATLAB 提供的基于模型化图形组态的控制

系统仿真软件，其命名直观地表明了该软件所具有的 SIMU（仿真）与 LINK（连接）两大功能，它使得一个复杂的控制系统的数字仿真问题变得十分直观而且相当容易。例如，对于图 1-13 所示的高阶 PID 控制系统，采用 Simulink 实现的仿真界面如图 1-14 所示。

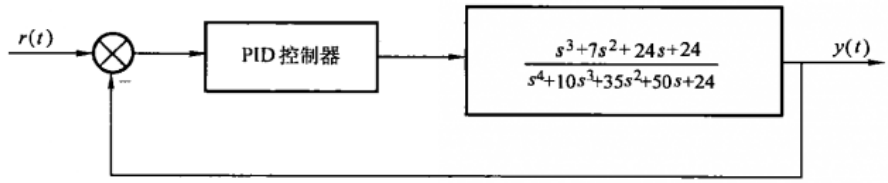


图 1-13 PID 控制系统结构图

值得一提的是，图 1-14 所示的仿真实现过程全部是在鼠标与键盘下完成的，从模型生成、参数设定到仿真结果的产生不过几分钟的时间，即使再复杂一些的系统仿真问题所需的时间也不会太多。Simulink 使控制系统数字仿真与 CAD 技术进入到人们期盼已久的崭新阶段。

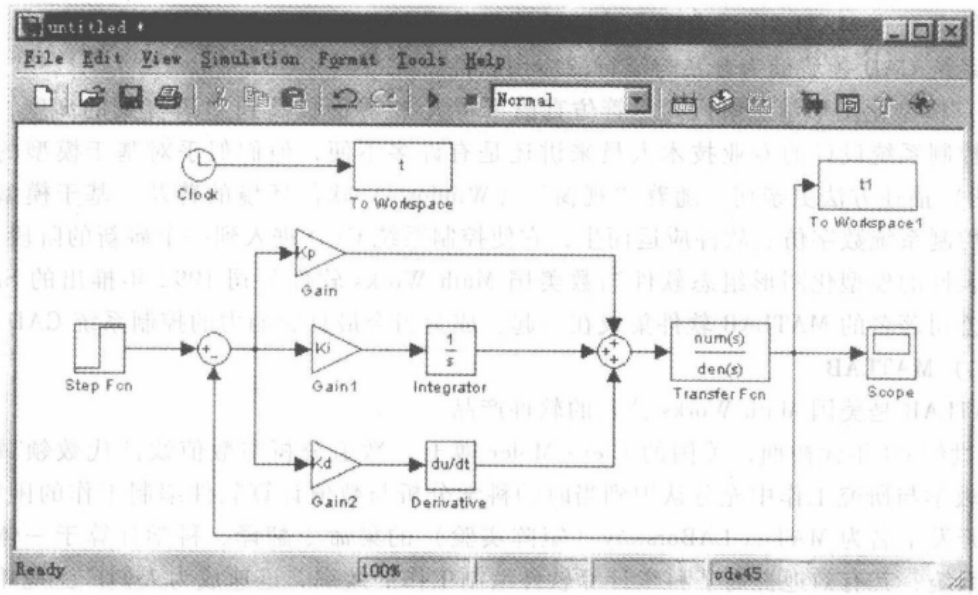


图 1-14 PID 控制系统的 Simulink 实现

（四）几种常用的仿真工具软件

目前，在自动化领域内应用的仿真软件较多。除了前面介绍的 MATLAB 软件以外，还有如下几种常用的仿真工具软件。

（1）ADAMS 机械系统动力学自动分析软件 ADAMS（Automatic Dynamic Analysis of Mechanical Systems），是美国 MDI 公司（Mechanical Dynamics Inc.）开发的著名“虚拟样机分析软件”（其后来被美国著名仿真分析软件公司 MSC 收购）。

目前，ADAMS 已经被全世界各行各业的数百家主要制造商采用。根据 1999 年机械系统动态仿真分析软件国际市场份额的统计资料，ADAMS 软件占据了销售总额近 8 千万美元中 51% 的份额。

ADAMS 一方面是虚拟样机分析的应用软件，用户可以运用该软件非常方便地对虚拟机械系统进行静力学、运动学和动力学分析；另一方面，又是虚拟样机分析开发工具，其开放性的程序结构和多种接口，可以成为特殊行业用户进行特殊类型虚拟样机分析的二次开发工

具平台。

ADAMS 的进一步了解请登录网站：www. mscsoftware. com. cn。

（2）Saber Saber 是美国 Analogy 公司开发并于 1987 年推出的模拟及混合信号仿真软件；Saber 曾几易其主，2002 年新思公司并购了当时拥有 Saber 的 Avant! 公司，将其收入囊中。

作为一种系统级仿真软件，Saber 拥有先进的原理图输入、数据可视化工具、大型混合信号、混合技术模型库以及强大的建模语言和工具组合功能，可以满足用户多种复杂的仿真需求。

此外，Saber 拥有大型的电气、混合信号、混合技术模型库，能够满足机电一体化和电源设计的需求。该模型库向用户提供不同层次的模型，支持自上而下或自下而上的系统仿真方法。

与传统仿真软件不同，Saber 在结构上采用 MAST 硬件描述语言和单内核混合仿真方案，并对仿真算法进行了改进，使仿真速度更快、更有效。Saber 可以同时对模拟信号、事件驱动模拟信号、数字信号以及模数混合信号设备进行仿真。对于包含有 Verilog 或 VHDL 编写的模型进行仿真设计。

Saber 能够与通用的数字仿真器相连接，包括 Cadence 的 Verilog-XL、Model Technology 的 ModelSim 和 ModelSim Plus、Innoveda 的 Fusion 仿真器。由于 MATLAB 软件的仿真工具 Simulink 在软件算法方面有优势，而 Saber 在硬件方面出色，将两者集成为 Saber-Simulink，进行协同仿真。

Saber 可以分析从 SOC 到大型系统之间的设计，包括模拟电路、数字电路及混合电路。它通过直观的图形化用户界面全面控制仿真过程，并通过对稳态、时域、频域、统计、可靠性及控制等方面的分析来检验系统性能。Saber 产品被广泛应用于航空/航天、船舶、电气/电力电子、汽车等设计制造领域。

在电源和机电一体化设计方面，Saber 是主流的系统级仿真工具。

Saber 的进一步了解请登录网站：www. synopsys. com。

（3）PSPICE 用于模拟电路仿真的 SPICE（Simulation Program with Integrated Circuit Emphasis）软件于 1972 年由美国加州大学伯克利分校的计算机辅助设计小组利用 FORTRAN 语言开发而成，主要用于大规模集成电路的计算机辅助设计。

SPICE 的正式实用版 SPICE 2G 在 1975 年正式推出，但是该程序的运行环境至少为小型机。1985 年，加州大学伯克利分校用 C 语言对 SPICE 软件进行了改写，1988 年 SPICE 被定为美国国家工业标准。与此同时，各种以 SPICE 为核心的商用模拟电路仿真软件，在 SPICE 的基础上做了大量实用化工作，从而使 SPICE 成为最为流行的电子电路仿真软件。

PSPICE 则是由美国 Microsim 公司在 SPICE 2G 版本的基础上升级并用于 PC 上的 SPICE 版本，其中采用自由格式语言的 5.0 版本自 20 世纪 80 年代以来在我国得到广泛应用，并且从 6.0 版本开始引入图形界面。1998 年著名的 EDA 商业软件开发商 OrCAD 公司与 Microsim 公司正式合并，自此 Microsim 公司的 PSPICE 产品正式并入 OrCAD 公司的商业 EDA 系统中。目前，OrCAD 公司已正式推出了 OrCAD PSPICE Release 9.0。

与传统的 SPICE 软件相比，PSPICE 9.0 在三大方面实现了重大变革：①在对模拟电路进行直流、交流和瞬态等基本电路特性分析的基础上，实现了蒙特卡罗分析、最坏情况分析

以及优化设计等较为复杂的电路特性分析；②不但能够对模拟电路进行仿真，而且能够对数字电路、数/模混合电路进行仿真；③集成度大大提高，电路图绘制完成后可直接进行电路仿真，并且可以随时分析观察仿真结果。

虽然 PSPICE 应用越来越广泛，但是也存在着明显的缺点。由于 SPICE 软件原先主要是针对信息电子电路设计而开发的，因此器件的模型都是针对小功率电子器件的，对于电力电子电路中所用的大功率器件存在的高电压、大注入现象不尽适用，有时甚至可能导致错误的结果。PSPICE 采用变步长算法，对于以周期性的开关状态变化的电力电子电路而言，大量的时间将耗费在寻求合适的步长上，从而导致计算时间的延长，有时甚至不收敛。另外，在磁性元件的模型方面 PSPICE 也有待加强。

PSPICE 的进一步了解请登录网站：www. orcad. com。

（4）ANSYS　ANSYS 是融结构、热、流体、电磁和声学于一体的大型通用有限元分析软件，对于求解热-结构耦合、磁-结构耦合以及电-磁-流体-热耦合等多物理场耦合问题具有其他软件不可比的优势。该软件可用于固体力学、流体力学、传热分析以及工程力学和精密机械设计等多学科的计算。ANSYS 软件主要包括三个部分：前处理模块、分析计算模块和后处理模块。

前处理模块提供了一个强大的实体建模及网格划分工具，用户可以方便地构造有限元模型；分析计算模块包括结构分析（可进行线性分析、非线性分析和高度非线性分析）、流体动力学分析、电磁场分析、声场分析、压电分析以及多物理场的耦合分析，可模拟多种物理介质的相互作用，具有灵敏度分析及优化分析能力；后处理模块可将计算结果以彩色等值线显示、梯度显示、矢量显示、粒子流迹显示、立体切片显示、透明及半透明显示（可看到结构内部）等图形方式显示出来，也可将计算结果以图表、曲线形式显示或输出。

软件提供了 100 种以上的单元类型，用来模拟工程中的各种结构和材料。该软件有多种不同版本，可以运行在从个人机到大型机的多种计算机设备上，如 PC、SGI、HP、SUN、DEC、IBM、CRAY 等。

ANSYS 的进一步了解请登录网站：www. ansys. com. cn。

（5）MSC. Nastran　MSC. Software 公司自 1963 年开始从事计算机辅助工程领域 CAE 产品的开发和研究。在 1966 年，美国国家航空航天局（NASA）为了满足当时航空航天工业对结构分析的迫切需求，招标开发大型有限元应用程序，MSC. Software 一举中标，负责了整个 Nastran 的开发过程。经过 40 多年的发展，MSC. Nastran 已成为 MSC 倡导的虚拟产品开发（VPD）整体环境最主要的核心产品，MSC. Nastran 与 MSC 的全系列 CAE 软件进行了有机的集成，为用户提供功能全面、多学科集成的 VPD 解决方案。

MSC. Nastran 是 MSC. Software 公司的旗舰产品，经过 40 余年的发展，用户从最初的航空航天领域，逐步发展到国防、汽车、造船、机械制造、兵器、铁道、电子、石化、能源、材料工程、科研教育等各个领域，成为用户群最多、应用最为广泛的有限元分析软件。

MSC. Nastran 的开发环境通过了 ISO 9001：2000 的论证，MSC. Nastran 始终作为美国联邦航空管理局（FAA）飞行器适航证领取的唯一验证软件。在中国，MSC 的 MCAE 产品作为与钢制压力容器分析设计标准 JB 4732—1995 相适应的设计分析软件，全面通过了全国压力容器标准化技术委员会的严格考核认证。另外，MSC. Nastran 是中国船级社指定的船舶分析验证软件。

MSC. Nastran 是功能强大、应用最为广泛、最为通用的结构有限元分析软件，可以进行结构强度、刚度、动力、随机振动、频谱响应、热传导、非线性、转子动力学、参数及拓扑优化、气动弹性等全面的仿真分析，是公认的业界标准。

MSC. Nastran 的进一步了解请登录网站：http：//www. mscsoftware. com. cn/MscProducts/productdetail. asp? id = 237。

（6）MSC. PATRAN　MSC. PATRAN 最早是由美国国家航空航天局（NASA）倡导开发的，是工业领域最著名的并行框架式有限元前后处理及分析系统，其开放式、多功能的体系结构可将工程设计、工程分析、结果评估、用户化身和交互图形界面集于一身，构成一个完整的 CAE 集成环境。

并行 CAE 工程的设计思想使 MSC. PATRAN 从另一个角度上打破了传统有限元分析的前后处理模式，其独有的几何模型直接访问技术（Direct Geometry Access，DGA）为基础的 CAD/CAM 软件系统间的几何模型沟通及各类分析模型无缝连接提供了完美的集成环境。使用 DGA 技术，应用工程师可直接在 MSC. PATRAN 框架内访问现有 CAD/CAM 系统数据库，读取、转换、修改和操作正在设计的几何模型而无需复制。MSC. PATRAN 支持不同的几何传输标准，包括 Parasolid、ACIS、STEP、IGES 等格式。

有限元分析模型可从 CAD 几何模型上快速地直接生成，用精确表现真实产品设计取代以往的近似描述，进而省去了在分析软件系统中重新构造几何模型的传统过程，MSC. PATRAN 所生成的分析模型（包含直接分配到 CAD 几何上的载荷、边界条件、材料和单元特性）将驻留 PATRAN 的数据库中，而 CAD 几何模型将继续保存在原有的 CAD/CAM 系统中，当相关的设计模型存储在 MSC. PATRAN 中并生成有限元网格时，原有的设计模型将被“标记”。设计与分析之间的相关性可使用户在 MSC. PATRAN 中迅速获知几何模型的任何改变，并能重新观察新的几何模型确保分析的精度。

MSC. PATRAN 的进一步了解请登录网站：www. mscsoftware. com/products/products _ detail. cfm? PI = 6。

（7）MSC. Marc 工具软件　MSC. Marc 是 MSC. Software 公司于 1999 年收购的 MARC 公司的产品。MARC 公司始创于 1967 年，是全球首家非线性有限元软件公司。经过 30 余年的不懈努力，Marc 软件得到学术界和工业界的大力推崇和广泛应用，建立了它在全球非线性有限元软件行业的领导者地位。

随着 Marc 软件功能的不断扩展，其应用领域也从开发初期的核电行业迅速扩展到航空、航天、汽车、造船、铁道、石油化工、能源、电子元件、机械制造、材料工程、土木建筑、医疗器材、冶金工艺和家用电器等，成为许多知名公司和研究机构研发新产品和新技术的必备工具。

Marc 软件通过了 ISO 9001 质量认证。在中国，Marc 通过了全国压力容器标准化技术委员会的严格考核和认证，成为与钢制压力容器分析设计标准 JB 4732—1995 相适应的有限元分析软件。

MSC. Marc 软件是功能齐全的高级非线性分析软件，具有极强的结构分析能力，可以处理各种复杂的非线性问题——几何非线性（大变形和大应变）、材料非线性和接触非线性，其高效的并行计算能力能够实现超大模型的非线性计算。

MSC. Marc 的进一步了解请登录网站：http：//www. mscsoftware. com. cn/MscProducts/

productdetail. asp? id = 222。

第五节 仿真技术的应用与发展

现代仿真技术经过近 50 年的发展与完善，已经在各行业做出卓越贡献，同时也充分体现出其在科技发展与社会进步中的重要作用。

一、仿真技术在工程中的应用

（一）航空航天工业

对于航空航天工业的产品来说，系统的庞杂，造价的高昂等因素促成了其必须建立起完备的仿真实验体系。在美国，1958 年所进行的四次发射全部失败了，1959 年的发射成功率也不过 57%；通过对实际经验的不断总结，美国国家航空航天局逐步建立了一整套仿真实验体系，到了 20 世纪 60 年代成功率达到 79%，在 70 年代已达到 91%，近年来，其空间发射计划已很少有不成功的情况了。

英法两国合作生产的“协和式”飞机，由于采用了仿真技术，使研制周期缩短了 1/8 ~ 1/6，节省经费 15% ~25%。

目前，我国及世界各主要发达国家的航空航天工业均相继建立了大型仿真实验机构，并形成了三级仿真实验体系，如图 1-15 所示，以保证飞行器从设计到定型生产过程的经济性与安全性。

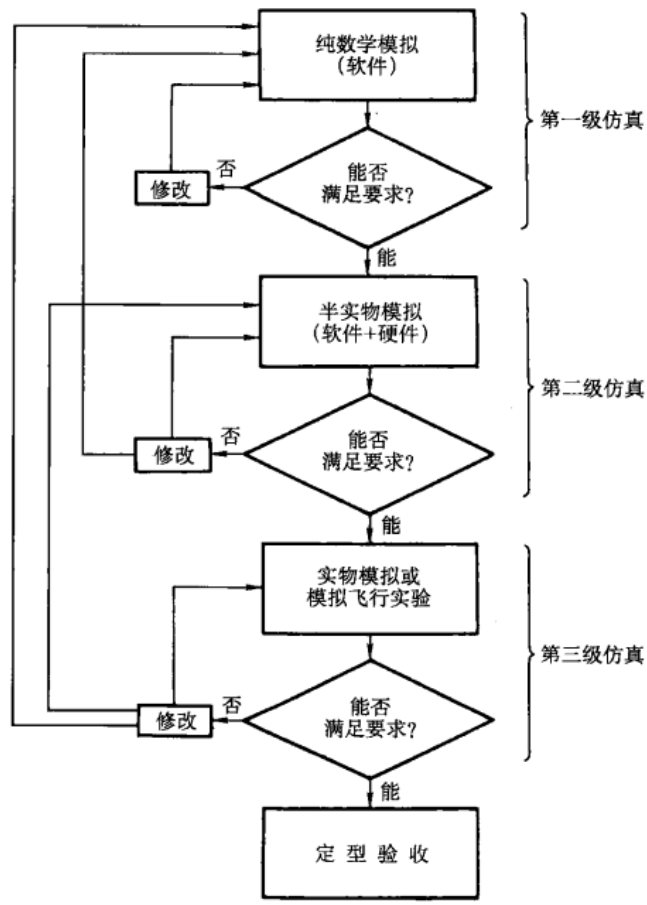


图 1-15 飞行器设计的三级仿真体系

此外，近年来在飞行员及宇航员训练用飞行仿真模拟器方面相继研制出多种产品，它主要包括：计算机系统、六自由度运动系统、视景系统（计算机成像）等设备，收到了方便、经济、安全的效果。

（二）电力工业

电力系统是最早采用仿真技术的领域之一。在电力系统负荷分配、瞬态稳定性以及最优潮流等方面，国内较早地采用了数字仿真技术，取得显著的经济效益。在三峡水利工程的子项目——大坝排沙系统工程设计中，设计人员也采用了物理仿真的方法，取得了较完善的研究成果。

近年来，国内在电站操作人员

培训模拟系统的研制上，达到国际先进水平，为仿真技术的应用开辟了广阔的前景。

（三）原子能工业

由于能源的日趋紧张，原子能的和平利用在世界范围内为人们广泛重视。随着核反应堆的尺寸与功率的不断增加，使得整个原子能电站运行的稳定性、安全性与可靠性等问题成为必须要解决的首要问题。因此，几乎大部分核电站都建有相应的仿真系统，许多仿真器是全尺寸的，即仿真系统与真实系统是完全一致的，只是对象部分，如反应堆、涡轮发电机及有关的动力装置则是用计算机来模拟的。核电站仿真器用来训练操作人员以及研究异常故障的排除处理，对于保证系统的安全运行是十分重要的。

目前，我国及世界各主要核技术先进国家在这方面均建立了相当规模的仿真实验体系，并取得了可观的成果。

（四）石油、化工及冶金工业

石油、化工生产过程中有一个显著的特点就是过程缓慢，而且往往过程控制、生产管理、生产计划、经济核算等搅在一起，使得综合效益指标难于预测与控制。因此，仿真实验成为石油、化工及冶金系统设计与分析研究的基本手段，仿真技术对这些领域的技术进步也不同程度地起到了促进作用。

（五）非工程领域

（1）医学　仿真技术在病变模型的建立、治疗方案的寻优、化疗与电疗强度的选择以及最佳照射条件等方面的应用，可为患者节省不必要的损失，为医生提供参考依据。

（2）社会学　在人口增长、环境污染、能源消耗以及病情防疫等方面，利用仿真技术可有效解决预测与控制问题。如我国人口模型的建立与研究，预测了未来100年我国人口发展的趋势，从而为计划生育控制策略的提出以及相关问题的解决起到了重要作用。此外，工业化、人口、环境这三个人类发展不容回避的问题日益引起人们的关注，如何建立相互制约的关系体制，走出一条可持续发展的良性循环的道路是近年来人们应用仿真技术进行研究的热点之一。

（3）宏观经济与商业策略的研究　随着人类经济发展的多元化与商业贸易的复杂化，在金融、证券、期货以及国家宏观经济调整等方面，数字仿真技术已成为不可缺少的有力工具。

二、应用仿真技术的重要意义

由于仿真技术具有经济、安全、快捷的优点以及其特殊的用途，使得其在工程设计、理论研究、产品开发等方面具有重要意义。

（一）仿真技术的优点

（1）经济　对于大型复杂系统，直接实验的费用往往是十分昂贵的，如空间飞行器一次飞行实验的成本大约为10^8美元，而采用仿真实验方法仅需其成本的1/10～1/5，而且设备可以重复使用。这类例子很多，读者不妨自己想一想。

（2）安全　对于某些系统（如载人飞行器、核电装置等），直接实验往往存在很大危险，甚至是不允许的；而采用仿真实验可以有效降低危险程度，对系统的研究起到保障作用。

（3）快捷　在系统分析与设计、产品前期开发以及新理论的检验等方面，采用仿真技

术（或 CAD 技术）可使工作进程大大加快，在科技飞速发展与市场竞争日趋激烈的今天，这一点是非常重要的。例如，现代服装设计采用仿真与 CAD 技术，使得设计师能够在多媒体计算机上实现不同身材、不同光照、不同色彩以及不同风向条件下所设计时装各种情况的展示，极大地促进了时装业的创新，有利于企业能够在激烈的市场竞争中处于不败之地。

（二）仿真技术的特殊功能

应用仿真技术可实现“预测”、“优化”等特殊功能。

（1）优化设计　在真实系统上进行结构与参数的优化设计是非常困难的，有时甚至是不可能的。在仿真技术中应用各种最优化原理与方法实现系统的优化设计，可使最终结果达到“最佳”，其对于大型复杂系统问题的研究具有重要意义。

（2）预测　对于一类非工程系统（如社会、经济、管理等系统），由于其规模及复杂程度巨大，直接进行某种实验几乎是不可能的，为减少错误的方针策略在以后的实践中所带来的不必要的损失，可以应用仿真技术对所研究系统的特性及其对外界环境的影响等问题进行“预测”，从而取得“超前”的认识，对所研究的系统实施有效的控制。

仿真与 CAD 技术对科技进步与产业发展有着不可估量的作用和意义，我们对它应予以足够的重视。

三、仿真技术的发展趋势

1）在硬件方面，基于多 CPU 并行处理技术的全数字仿真系统将有效提高仿真系统的速度，从而使仿真系统的“实时性”得以进一步加强。

2）随着网络技术的不断完善与提高，分布式数字仿真系统将为人们广泛采用，从而达到“投资少、效果好”的目的。

3）在应用软件方面，直接面向用户的高效能的数字仿真软件将不断推陈出新，各种专家系统与智能化技术将更深入地应用于仿真软件开发中，使得在人-机界面、结果输出、综合评判等方面达到更理想的境界。

4）虚拟现实技术的不断完善，为控制系统数字仿真与 CAD 开辟了一个新时代。例如，在飞行器驾驶人员培训模拟仿真系统中，可采用虚拟现实技术，使被培训人员置身于模拟系统中就犹如身在真实环境里一样，使得培训效果达到最佳。

虚拟现实技术是一种综合了计算机图形技术、多媒体技术、传感器技术、显示技术以及仿真技术等多种学科而发展起来的高新技术。

5）随着 FMS 与 CIMS 技术的应用与发展，“离散事件系统”越来越多地为仿真领域所重视，离散事件仿真从理论到实现给我们带来了许多新的问题。随着管理科学、柔性制造系统、计算机集成制造系统的不断发展，“离散事件系统仿真”问题将越来越显示出它的重要性。

第六节　问题与探究——虚拟现实与仿真技术

一、虚拟现实技术

（一）概述

当代科学技术的发展为适应信息社会发展的需要，要求我们提高人与信息社会的交互能力，以提高人对信息的理解能力。现今，人们不仅要求通过书面字意去观察信息处理的结果，还要求能通过人们的视觉、听觉、触觉以及形体、手势或者口令等信息，参与到信息处理的环境中去，从而获得身临其境的体验。这种信息处理系统已不再是建立在一个单维的数字化信息空间上，而是建立在一个多维化的信息空间之中，建立在一个定性与定量相结合、感性认识与理性认识相结合的综合集成环境中。虚拟现实技术就是支撑这个多维信息空间的关键技术。

虚拟现实（简称 VR），又称灵境技术，是以沉浸性、交互性和构想性为基本特征的计算机高级人机界面。如图 1-16 所示，它综合利用了计算机图形学、仿真技术、多媒体技术、人工智能技术、计算机网络技术、并行处理技术和多传感器技术，模拟人的视觉、听觉、触觉等感觉器官功能，使人能够沉浸在计算机生成的虚拟境界中，并能够通过语言、手势等自然的方式与之进行实时交互，创建了一种适人化的多维信息空间。使用者不仅能够通过虚拟现实系统感受到在客观物理世界中所经历的“身临其境”的逼真性，而且能够突破空间、时间以及其他客观限制，感受到真实世界中无法亲身经历的体验。

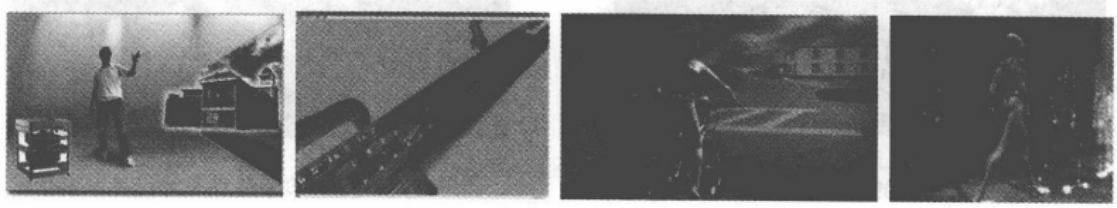


图 1-16 虚拟现实技术

目前，虚拟现实技术已经和理论分析、科学实验一起，成为人类探索客观世界规律的三大手段。

（二）虚拟现实的特征与分类

（1）虚拟现实的特征 美国学者伯第亚（G. Burdea）在“Virtual Reality Technology”一文中提出，虚拟现实具有最重要的三个特征：Interaction（交互）、Immersion（沉浸）和 Imagination（想象），也就是人们熟称的 3I 特征。用“虚拟现实三角形”来说明虚拟现实技术的基本特征，如图 1-17 所示。其中，“交互性”是指参与者通过使用专业设备，用人类的自然技能实现对模拟环境内物体的可操作程度和从环境中得到反馈的自然程度（包括实时性）；“沉浸感”是 VR 最主要的技术特征，它力图使用户在计算机所创建的三维虚拟环境中处于一种身临其境的感觉，觉得自己是虚拟环境的一部分，而不是旁观者。VR 技术并不只是一种媒介或一个高层终端与用户见面，它的应用能解决在工程、医学、军事等方面的很多问题。而这些应用系统的设计与开发就极大地依赖于人类的想象能力，即 VR 的第三个特征——想象。

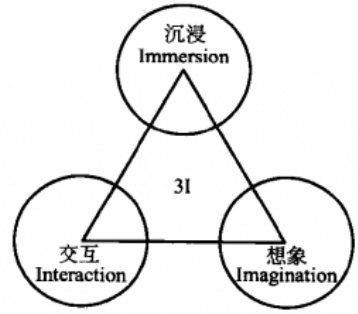


图 1-17 虚拟现实的 3I 特征

（2）虚拟现实的分类 虚拟现实可分为三类形式：仿真性、超越性、幻想性。

1）仿真性的虚拟现实。仿真性的虚拟现实是根据现实世界的真实物理法则，由计算机

将其模拟出来。它虽然是一种模拟环境，但一切在数学上都是符合客观规律的。作为仿真性的虚拟现实，其类似于人的思维通过虚拟现实技术实现的观念再造并使之形象化。图 1-18 所示即为该领域最为常见的例子——发动机的虚拟现实，它能够为发动机的设计者提供真实可靠的仿真数据，以使其能够快速且持续不断地进行优化完善。

2）超越性的虚拟现实。超越性的虚拟现实虽然也是根据真实的物理法则进行模拟，但所模拟的对象或者用人的五官无法感觉到，或者在日常生活中无法接触到。作为超越性的虚拟，虚拟现实可以充分发挥人们的认识和探索能力，为人类探索未知世界的奥秘提供服务。图 1-19 所示即为虚拟星云系统，它是超越我们所能感知的世界而存在的，仅能够通过虚拟现实技术来还原实现。

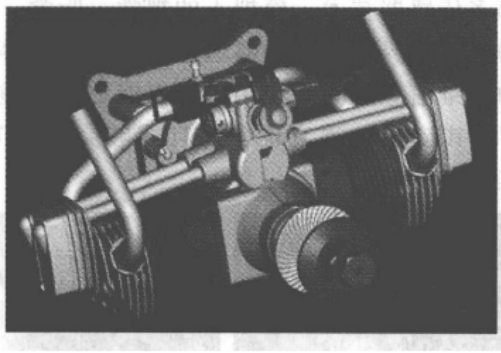


图 1-18　仿真性的虚拟现实——发动机

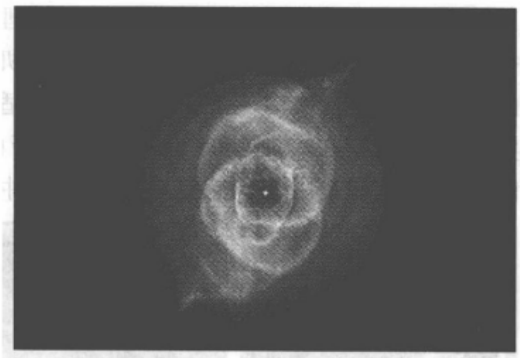


图 1-19　超越性的虚拟现实——星云系统

3）幻想性的虚拟现实。幻想性的虚拟现实可以根本无视客观的物理法则，完全把凭空想象出来的东西，用计算机图像、音响等功能，将其变成可以看到、听见的多媒体作品。因此，作为幻想性的虚拟，它给人们带来广阔的想象时空，尽管有时是荒诞不经的，但也促进了人类想象和创造力的发展，图 1-20 所示为电子游戏中出现的幻想性虚拟现实场景。



图 1-20　幻想性的虚拟现实——电子游戏中的场景

总之，虚拟现实现已成为目前用途最为广泛、影响最为深远的多媒体仿真技术。它使人们能想象一个曾见过的现实世界，也能想象一个未曾见过的虚幻世界。人们在虚拟现实中体验和认识到的，可能是真实世界的仿真，也可能是一个抽象概念的形象，也可能是一些并不符合真实物理规律，有悖人们常识的某种稀奇古怪的幻想。同时，虚拟现实作为一种前景广阔的网络文化现象，也将会给人们提出许多新的值得探讨的课题。

（三）虚拟现实技术的应用领域

因为虚拟现实技术的特点，所以对人类社会的意义是非常大的。它和其他很多信息技术一样，当信息技术领域的专家还未把它的理论和技术探讨得十分清楚时，它已渗透到科学、技术、工程、医学、文化、娱乐的各个领域了，受到各个领域人们的高度重视。

（1）军事领域　美国国防部和军方认为：虚拟现实将在武器系统性能评价、武器操纵

训练及指挥大规模军事演习三方面发挥重大作用。他们制定了战争综合演示厅计划、防务仿真交互网络计划、综合战役计划以及虚拟座舱等应用环境，并在核武器试验及许多局部战争中进行了应用。图 1-21 所示为基于虚拟现实技术的军事演习场景。



图 1-21　虚拟军事演习

（2）航天航空　美国国家航空航天局是虚拟现实最早的研究单位和应用者。宇宙飞船及各类航空器是需耗费巨资的现代化工具，而进入宇宙有大量未知、危险的因素，因而模拟各种航空器可能遇到的环境，不仅可节省大量费用，而且是十分必要的。图 1-22 所示的虚拟风洞就是一例。

（3）计算机辅助设计　各种工业产品、建筑物均需反复构思和设计，但往往用户仍不满意。美国波音公司的巴特勒（Butler）设计了一架名为 VS-X 的虚拟飞机，它可使设计人员有身临其境观察飞机外形、内部结构及布局的效果。同样，建筑设计师可在盖楼前通过虚拟建筑物，让用户自己来观察外形和内部房间部位，也便于设计师修改设计。图 1-23 所示为虚拟现实技术在计算机辅助设计中的若干应用，包括结构设计与分析、强度校核、方案优化等。

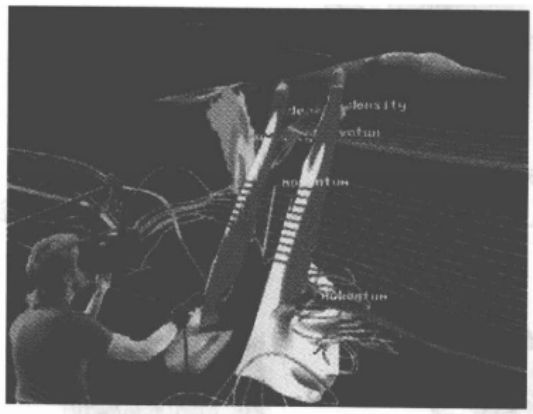


图 1-22　虚拟风洞

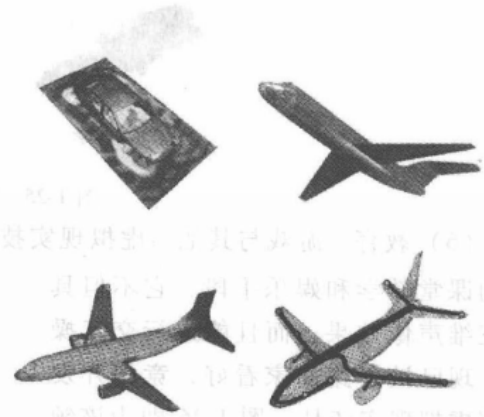


图 1-23　虚拟现实技术在计算机辅助设计中的应用

（4）外科手术和人体器官的模拟　外科医生的培训是一项投资大、时间较长的工作，这是因为不能随便让实习医生在病人身上动手术，可是不亲自动手，又如何学会手术呢？虚拟手术台已能部分模仿外科手术的现场。同样，提供模拟的人体器官，可让学生逼真地观察器官内部的构造和病灶，具有极高的实验价值。图 1-24 所示即为利用虚拟现实技术所展现的人体器官心脏及其周边血管。

（5）科学研究与计算的可视化　现代工程设计与分析中，各种分子结构模型、大坝应力计算的结果、地震石油勘探数据处理等，均需要三维（甚至多维）图形可视化的显示和交互浏览，虚拟样机技术等；虚拟现实技术为科学研究、探索微观形态等提供了形象直观的工具。虚拟现实技术在科学研究中最直接的应用即为虚拟样机技术，如图 1-25 所示，该技术是多学科技术及相关软件综合应用的产物，它的应用范围极其广泛。

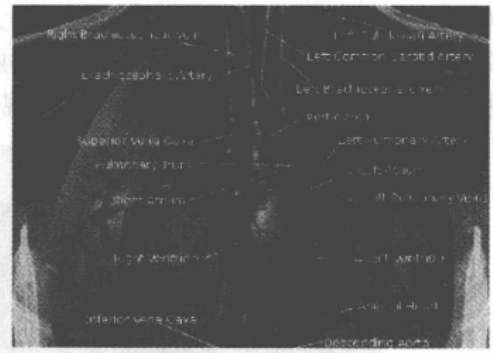


图 1-24　虚拟人体器官



图 1-25　虚拟样机技术应用

（6）教育、游戏与其他　虚拟现实技术向人们，尤其是为青少年提供了一种生动与深刻的课堂教学和娱乐手段，它不但具有三维声像效果，而且能进行交互操作，现已被各路商家看好，竞相开发高档虚拟现实产品。图 1-26 即为该领域的代表作——任天堂出品的 Wii 游戏机堪称是目前虚拟现实技术与娱乐最完美的结合。

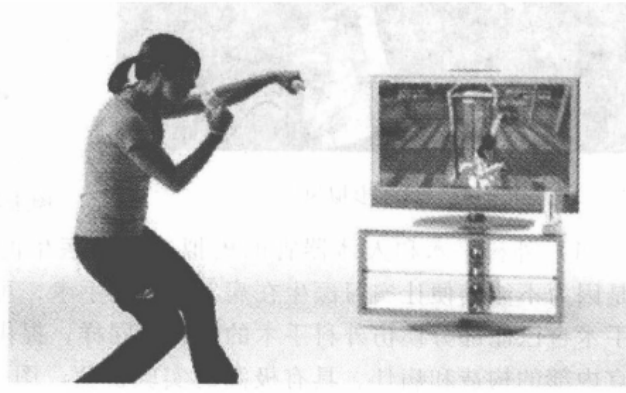


图 1-26　任天堂 Wii 动感游戏机

二、虚拟现实仿真技术

（一）虚拟样机技术

20 世纪 80 年代初，“虚拟现实”

这一概念一经推出，就在系统仿真领域中产生了巨大的影响。20 多年来，虚拟现实技术与仿真技术相结合使得系统仿真结果在表现形式上更加逼真、形象，使得人更富有想象力；而系统仿真技术在虚拟现实中的应用（如运动学、动力学等原理的应用）使得虚拟现实技术得以更广泛的应用（如影视特技、电脑游戏等）。下面是虚拟现实仿真技术在自动化领域中的几个典型应用。

（1）虚拟样机　现在，人们可以借助于虚拟现实仿真工具软件（如 ADAMS）来“制造”机械部件、设备、车辆甚至飞行器的“虚拟样机”，在计算机上对“样机”进行各种动态/静态性能的测试；目前，虚拟样机“制造”的准确性可达 95% 以上，可广泛应用于制造业和科研工作中。

（2）虚拟制造　在现代工业设计、机械设计等领域，人们可广泛应用虚拟现实工具软件（如 Pro/E、Auto CAD）进行机械部件或复杂机械装配机构的“虚拟制造”，在计算机上进行三维运动学实验与装配实验，以检查设计的有效性。

（3）虚拟环境　在现代艺术设计、影视艺术设计以及平面艺术设计等领域，人们也已广泛应用虚拟现实仿真工具软件（如 VRML、3DSMAX 等）进行虚拟环境与场景的设计，以有效增强作品的感染力（或仿真环境的真实性）。

虚拟样机技术是一种基于虚拟的数字化设计方法，是各领域 CAX/DFX 技术的发展和延伸。虚拟样机技术进一步融合先进的建模与仿真技术、现代信息技术、先进设计制造技术和现代管理技术，将这些技术应用于复杂产品全生命周期、全系统，并对它们进行综合管理。与传统产品设计技术相比，虚拟样机技术强调系统的观点、涉及产品全生命周期、支持对产品的全方位测试、分析与评估，强调不同领域的虚拟化的协同设计。

（二）基于虚拟样机的联合仿真技术

近年来，多学科联合仿真是整个 CAE（Computer Aided Engineering）行业发展的方向，因为只有对关键学科之间复杂交互作用的准确表述才能保证真实地模拟物理现象，即单个解决方案可以进行多学科分析，但不能考虑多学科之间的交互作用。如何通过使用一个模型实现不同学科之间的交互作用和耦合、优化不确定因素及利用 64 位高性能处理器提高计算速度是目前仿真领域亟待解决的问题。

基于多领域协同建模与协同仿真的虚拟样机技术很好地解决了这个问题。各专业领域的工程师在设计过程中可以共享同一个产品的虚拟样机，无需制作物理样机就能够随时对虚拟样机的整体特性进行反复的仿真测试，直到获得满意的设计结果。联合仿真技术的典型应用有：多体动力学与控制系统（如车辆控制）、结构与气动载荷（如飞行动力学分析）等。

目前较为通用和流行的实现多领域协同仿真的方式主要有以下三种：

（1）联合仿真式（Co-Simulation）　不同仿真软件在各自运行前进行数据耦合关系定义和建立连接，仿真开始后，耦合的仿真数据通过进程间通信（IPC）或者网络通信的方式实现双向交换和调用。

（2）模型转换式（Model Transfer）　其主要原理是将一个仿真软件的模型转化为另一个仿真软件支持的特定格式的包含模型信息的数据文件或者动态链接库文件（.dll），实现模型级别的协同仿真。

（3）求解器集成式（Solver Convergence）　其基本原理是实现两个不同仿真软件之间的求解器集成，在其中一个仿真软件中可以调用另一个仿真软件的求解器，从而完成协同仿真。

有些仿真软件综合应用了以上三种协同方式，比如 ADAMS 和 Simulink 之间支持联合仿真和模型转换的协同仿真方法，而 ADAMS 和 EASY5 之间则全部支持以上三种方法。联合仿真式现今较为通用，它能够使这些软件同步运行，随时可以观察仿真结果的变化，具有良好的交互性，其中应用最为广泛的就是 ADAMS/MATLAB 联合仿真技术。

（三）基于 ADAMS/MATLAB 的联合仿真技术

（1）技术描述　ADAMS 软件是实现在计算机上仿真分析复杂机械系统的运动性能的工具软件。通过建立系统的 ADAMS 模型，可得到一个近似的虚拟实物模型。我们通常使用的 MATLAB 仿真是建立在数学模型的基础上的，其动态性能都是经过计算得来的。使用 ADAMS 建立的系统模型，其仿真模型相当于“虚拟样机”，能够较真实模拟实物的真实运动状况。ADAMS/Control（控制模块）可以将 ADAMS 的机械系统模型与控制系统应用软件（如 MATLAB、EASY5 或者 MATRIX）连接起来，实现在控制系统软件环境下的交互式仿真，还可以在 ADAMS 中观察仿真结果。因此将两种软件联合起来使用，不仅可以验证动力学模型的准确性及控制策略的实用性，还可以得到更加接近实际情况的仿真结果，并进行快速分析，进一步改善控制策略的性能。

（2）仿真方法及其步骤　在使用 ADAMS/Control 模块以前，机械设计师和控制工程师使用不同的软件对同一概念设计进行重复建模，并且进行不同的设计验证和实验，然后制造物理样机。一旦出现问题，不管是机械系统的故障还是控制系统的故障，两方都要重新设计，其传统设计流程如图 1-27 所示。

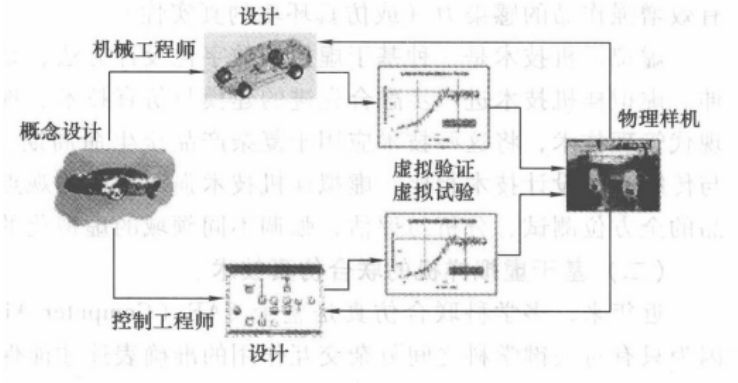


图 1-27　传统设计流程

使用 ADAMS/Control 模块后，机械设计师和控制工程师可以共享同一个虚拟模型，进行同样的设计验证和实验，使机械系统设计和控制系统设计能够协调一致，并且可以设计复杂模型，包括非线性模型和非刚体模型。既节约了设计时间，又增加了设计的可靠性，其设计流程如图 1-28 所示。

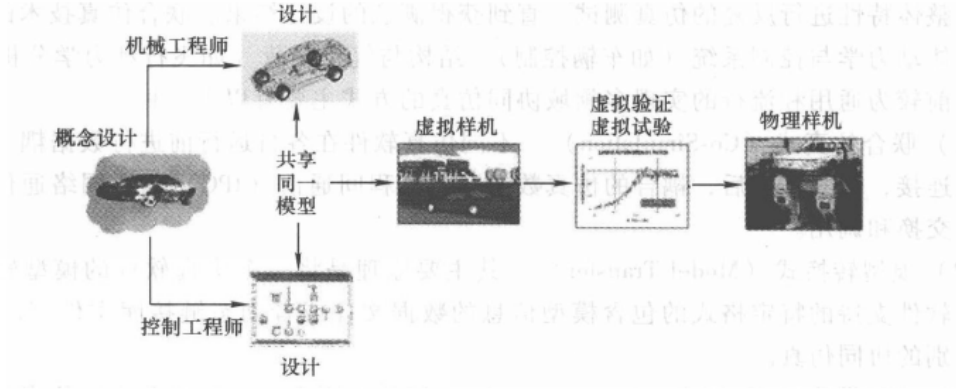


图 1-28　基于虚拟样机的设计流程

ADAMS/Control 控制系统设计主要有以下四个步骤，如图 1-29 所示。

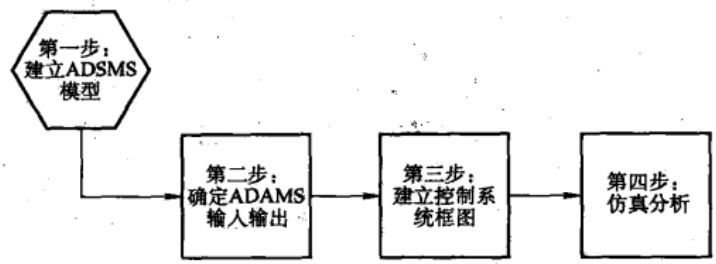


图 1-29 ADAMS/Control 设计流程

第一步：建立 ADAMS 模型。ADAMS 模型可以在 ADAMS/Control 下直接建立，也可以输入已经建好的外部模型。

第二步：确定 ADAMS 输入输出。通过确定 ADAMS 的输入和输出变量可以在 ADAMS和控制软件之间形成一个闭合回路，如图 1-30 所示。

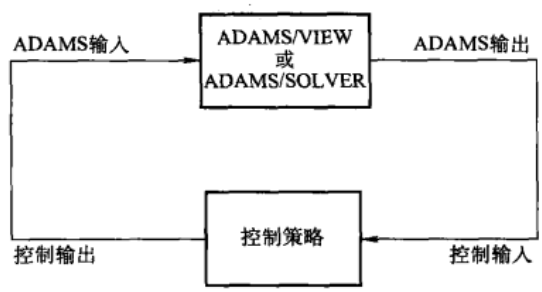


图 1-30 ADAMS 的输入和输出变量

第三步：建立控制系统框图。使用控制软件 MATLAB、EASY5 或者 MATRIX（建立控制系统模型）并将其与 ADAMS 机械系统模型连接起来。

第四步：仿真分析。可以使用交互式或批处理方式，建立机械系统和控制系统连接在一起的模型。

通过以上步骤，即可建立一套完整的基于 ADAMS/Simulink 联合仿真平台。

三、基于虚拟样机的球棒控制系统仿真

（一）问题提出

由刚性球和连杆臂构成的球棒系统如图 1-31 所示。连杆在驱动力矩 τ 的作用下绕轴心 O 点作旋转运动。连杆的转角和刚性球在连杆上的位置分别用 θ、L 表示，刚性球的半径为 R。当小球转动时，球的移动和棒的转动构成复合运动。

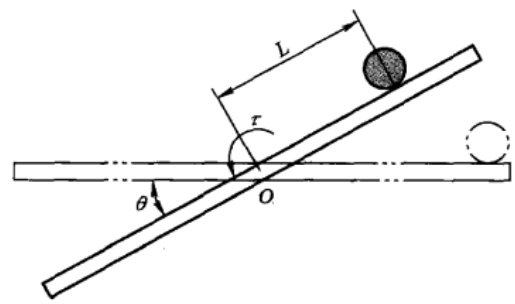


图 1-31 球棒系统结构图

球棒系统是一个典型的多变量非线性系统，是非线性控制理论的一个典型实验课题。该系统通过控制驱动电动机的力矩，以使刚性球稳定运动在连杆 L 上的任意位置。

（二）球棒系统的联合仿真实验

下面通过球棒系统的例子，来说明 ADAMS-MATLAB 动力学联合仿真的过程。

1）在 ADAMS 中建立球棒系统动力学模型，如图 1-32 所示。

2）定义状态变量，生成 ADAMS-MATLAB 模块。经过分析，定义系统的控制变量为转

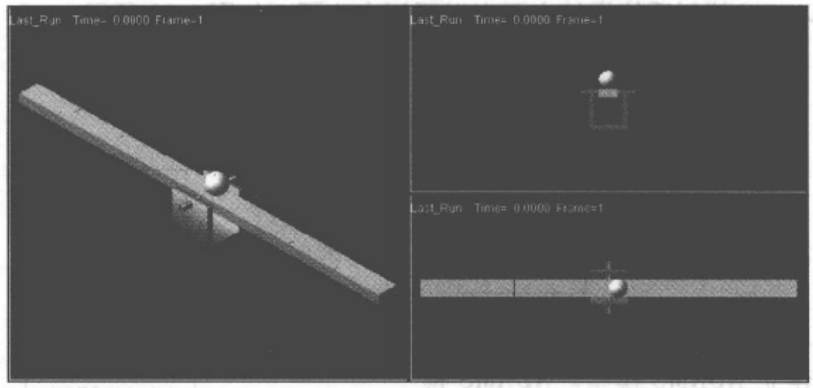


图 1-32 球棒系统的 ADAMS 模型

轴处的力矩，检测变量（控制目标）为小球在棒上的位置及棒与水平方向的转角。设置控制量、测量量为相应的状态变量，由 ADAMS 生成 MATLAB 模型文件，其组成如图 1-33、图 1-34 所示。

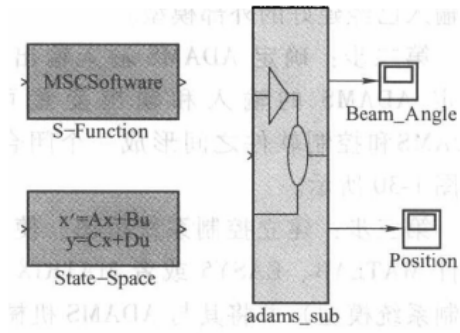


图 1-33 由 ADAMS 生成的 MATLAB 文件

打开 ADAMS Plant 模块可以看到一些仿真设置，其中的 Simulation mode（仿真模式）选择 discrete（离散）。Animation mode（动画模式）可选择 batch（批处理）：生成数据文件，不进行实时动画演示，待仿真完毕后，再导入 ADAMS Post Processor 中进行分析，其优点是仿真时间较短，对计算机系统要求不高；也可选 interactive（交互式）：实时进行动画演示，优点是仿真过程中对系统有更加直观的认识，并可以随时暂停仿真，缺点是对计算机系统，尤其是图形显示部件要求较高。

3）建立 MATLAB 控制系统。以 ADAMS-SUB 模块来代替 MATLAB 中的动力学建模部分，使用 Simulink 模块，运用相应的控制策略建立球棒系统控制框图。开始仿真后，如果选择的是交互式仿真模式，在屏幕上就可以看到动态的仿真过程，如图 1-35 所示。

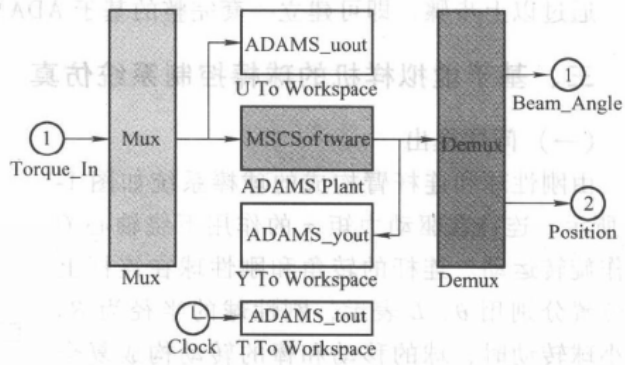


图 1-34 ADAMS-SUB 内部结构

通过以上步骤，就可以利用 MATLAB 强大的数值计算与控制模块来对任意一个在 ADAMS 中建立的“虚拟样机”进行系统的动力学仿真分析，这种方法可以获得大量直观的信息，可以帮助我们更好地运用仿真技术，实现对复杂机电系统的设计与研究。

（三）问题探究

建立球棒系统的虚拟样机模型，利用 ADAMS/MATLAB 动力学联合仿真方法，进行球棒系统的稳定控制，分析下列问题：

1）改变小球的材料（转动惯量），看一下控制效果又如何？控制策略应作如何改变？

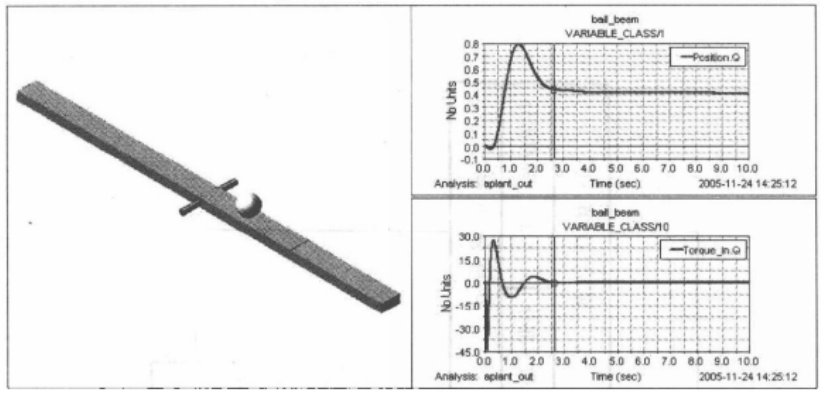


图 1-35 ADAMS-MATLAB 联合动力学仿真

2）改变长杆材料（转动惯量），看一下控制效果又如何？控制策略应作如何改变？

3）改变球与棒之间的动摩擦因数，看一下控制效果又如何？控制策略应作如何改变？

从中我们可以体会到：虚拟样机技术使得数字仿真过程中的参数修改、结构变更等问题变得快捷、系统、真实、有效。

小 结

本章就仿真技术所涉及的基本概念、发展状况等问题进行了概括的介绍，归纳起来有如下几点：

1）仿真是对系统进行研究的一种实验方法，数字仿真具有经济、安全、快捷的优点。

2）仿真实验所遵循的基本原则是相似原理。

3）仿真实验是在模型上进行的，建立系统模型是仿真实验中的关键内容，因为它直接影响仿真结果的真实性与可靠性。

4）系统模型分为物理模型、数学模型及描述模型，根据所采用模型的不同，可有实物仿真、半实物仿真、数字仿真等类型。

5）系统、模型与计算机是数字仿真的三个基本要素，建模、仿真与结果分析是其三项基本活动，在仿真实验中应充分重视建模与结果分析环节。

6）CAD 技术推动了设计领域的革命，是系统分析与设计的有力工具。

7）MATLAB 与 Simulink 是当今广泛采用的控制系统数字仿真与 CAD 应用软件，应熟练掌握之。

8）虚拟现实技术是一门综合性的交叉学科，具有更广阔的研究与发展前景，应给予充分重视。

习 题

1-1 什么是仿真？它所遵循的基本原则是什么？

1-2 在系统分析与设计中仿真法与解析法有何区别？各有什么特点？

1-3 数字仿真包括哪几个要素？其关系如何？

1-4 为什么说模拟仿真较数字仿真精度低？其优点如何？

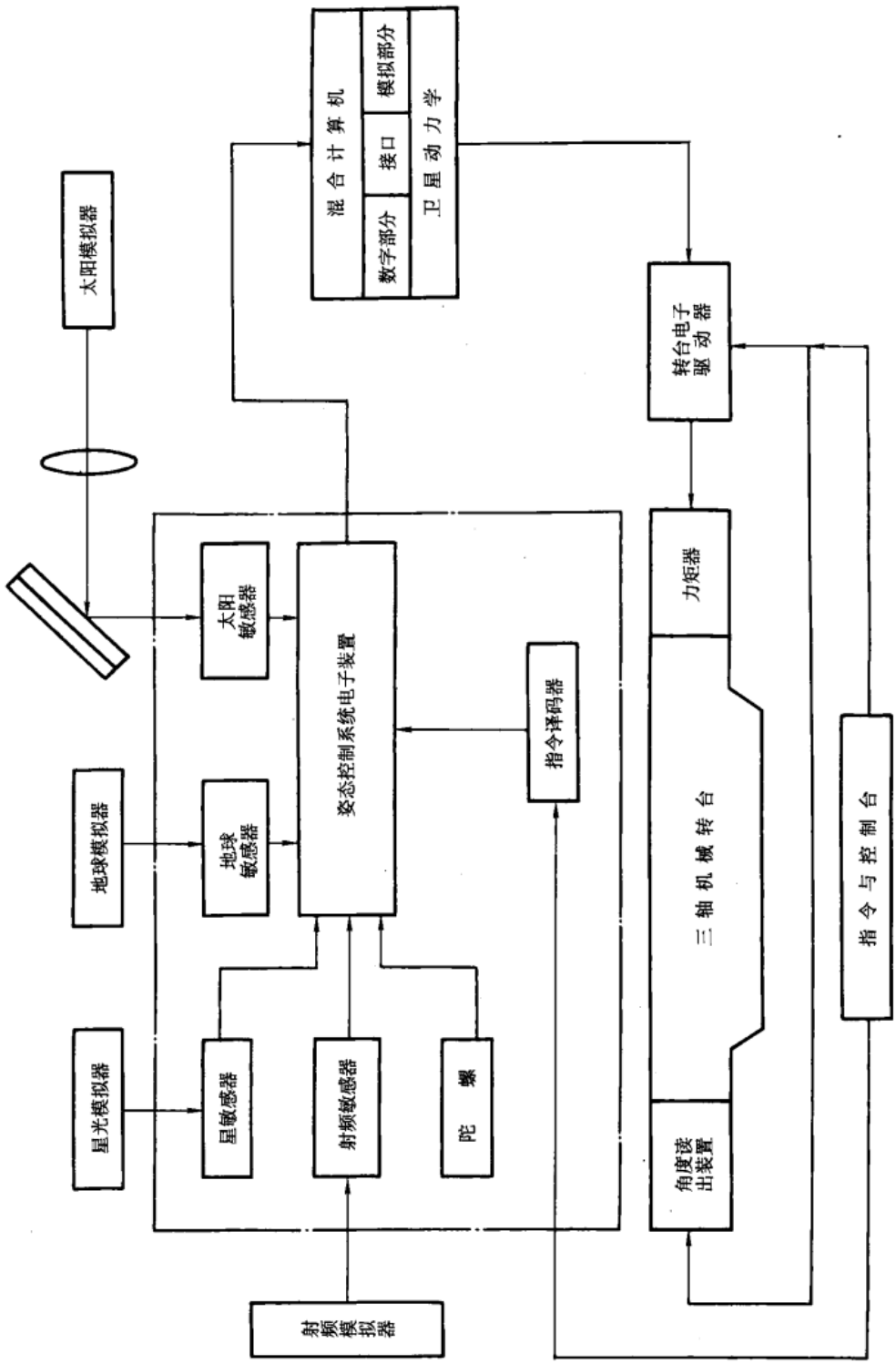


图 1-36 卫星姿态控制仿真实验系统

1-5 什么是 CAD 技术？控制系统 CAD 可解决哪些问题？

1-6 什么是虚拟现实技术？它与仿真技术的关系如何？

1-7 什么是离散系统？什么是离散事件系统？如何用数学的方法描述它们？

1-8 图 1-36 所示为某卫星姿态控制仿真实验系统，试说明：

1）若按模型分类，该系统属于哪一类仿真系统？

2）图中“混合计算机”部分在系统中起什么作用？

3）与数字仿真相比该系统有什么优缺点？

第二章　控制系统的数学描述

控制系统计算机仿真是建立在控制系统数学模型基础之上的一门技术。自动控制系统的种类繁多，为通过仿真手段进行分析和设计，首先需要用数学形式描述各类系统的运动规律，即建立它们的数学模型。模型确定之后，还必须寻求合理的求解数学模型的方法，即数值算法，才能得到正确的仿真结果。本章将从常见的控制系统数学模型的建立和模型之间的相互转换入手，引出计算机仿真的实现问题（二次模型化）和控制系统仿真常用的几种数值算法，并对数值求解常微分方程中应注意的重要概念，如误差与精度、计算步距、计算时间以及数值稳定性、病态问题等予以简明扼要的讨论和阐述。

第一节　控制系统的数学模型

工业生产中的实际系统绝大多数是物理系统，系统中的变量都是一些具体的物理量，如电压、电流、压力、温度、速度、位移等，这些物理量是随时间连续变化的，称之为连续系统；若系统中物理量是随时间断续变化的，如计算机控制、数字控制、采样控制等，则称为离散（或采样）系统。采用计算机仿真来分析和设计控制系统，首要问题是建立合理地描述系统中各物理量变化的动力学方程，并根据仿真需要，抽象为不同表达形式的系统数学模型。

一、控制系统数学模型的表示形式

根据系统数学描述方法的不同，可建立不同形式的系统数学模型。经典控制理论中，常用系统输入-输出的微分方程或传递函数表示各物理量之间的相互制约关系，被称为系统的外部描述或输入-输出描述；现代控制理论中，通过设定系统的内部状态变量，建立状态方程来表示各物理量之间的相互制约关系，这称为对系统的内部描述或状态描述。连续系统的数学模型通常可由高阶微分方程或一阶微分方程组的形式表示，而离散系统的数学模型是由高阶差分方程或一阶差分方程组的形式表示。如所建立的微分或差分方程为线性的，且各系数均为常数，则称之为线性定常系统的数学模型；如果方程中存在非线性变量，或方程中存在随时间变化的系数，则称之为非线性系统或时变系统数学模型。

本节主要讨论线性定常连续系统数学模型的几种表示形式。线性定常离散系统的数学模型将在后面章节中讨论。

（一）微分方程形式

设线性定常系统输入、输出量是单变量，分别为 $u(t)$、$y(t)$，则两者间的关系总可以描述为线性常系数高阶微分方程形式

$$a_0 y^{(n)} + a_1 y^{(n-1)} + \cdots + a_{n-1} y' + a_n y = b_0 u^{(m)} + \cdots + b_m u \tag{2-1}$$

式中，$y^{(j)}$ 为 $y(t)$ 的 j 阶导数，$y^{(j)} = \dfrac{\mathrm{d}^j y(t)}{\mathrm{d}t^j}$，$j=0$，$1$，$\cdots$，$n$；$u^{(i)}$ 为 $u(t)$ 的 i 阶导数，$u^{(i)} =$

$\frac{d^i u(t)}{dt^i}$，$i=0$，1，…，m；a_j 为 $y(t)$ 及其各阶导数的系数，$j=0$，1，…，n；b_i 为 $u(t)$ 及其各阶导数的系数，$i=0$，1，…，m；n 为系统输出变量导数的最高阶次；m 为系统输入变量导数的最高阶次，通常总有 $m \leqslant n$。

对式（2-1）的数学模型，可以用以下模型参数形式表征：

输出系统向量 $\boldsymbol{A}=[a_0, a_1, \cdots, a_n]$，$n+1$ 维

输入系数向量 $\boldsymbol{B}=[b_0, b_1, \cdots, b_m]$，$m+1$ 维

输出变量导数阶次，n

输入变量导数阶次，m

有了这样一组模型参数，就可以简便地表达出一个连续系统的微分方程形式。

微分方程模型是连续控制系统其他数学模型表达形式的基础，以下所要讨论的模型表达形式都是以此为基础发展而来的。

（二）状态方程形式

当控制系统输入、输出为多变量时，可用向量分别表示为 $\boldsymbol{U}(t)$、$\boldsymbol{Y}(t)$，由现代控制理论可知，总可以通过系统内部变量之间的转换设立状态向量 $\boldsymbol{X}(t)$，将系统表达为状态方程形式

$$\begin{cases} \dot{\boldsymbol{X}}(t) = \boldsymbol{A}\boldsymbol{X}(t) + \boldsymbol{B}\boldsymbol{U}(t) \\ \boldsymbol{Y}(t) = \boldsymbol{C}\boldsymbol{X}(t) + \boldsymbol{D}\boldsymbol{U}(t) \end{cases} \tag{2-2}$$

$$\boldsymbol{X}(t_0) = \boldsymbol{X}_0 \quad \text{为状态初始值}$$

已知，$\boldsymbol{U}(t)$ 为输入向量（m 维）；$\boldsymbol{Y}(t)$ 为输出向量（r 维）；$\boldsymbol{X}(t)$ 为状态向量（n 维）。因此，对式（2-2）的数学模型，则用以下模型参数来表示系统：

系统系数矩阵 $\boldsymbol{A}$（$n \times n$ 维）

系统输入矩阵 $\boldsymbol{B}$（$n \times m$ 维）

系统输出矩阵 $\boldsymbol{C}$（$r \times n$ 维）

直接传输矩阵 $\boldsymbol{D}$（$r \times m$ 维）

状态初始向量 $\boldsymbol{X}_0$（n 维）

简记为（$\boldsymbol{A}$，$\boldsymbol{B}$，$\boldsymbol{C}$，$\boldsymbol{D}$）形式。

应当指出，控制系统状态方程的表达形式不是唯一的。通常可根据不同的仿真分析要求而建立不同形式的状态方程，如能控标准型、能观标准型、约当型等。

（三）传递函数形式

将式（2-1）在零初始条件下，两边同时进行拉普拉斯变换，则有

$$(a_0 s^n + a_1 s^{n-1} + \cdots + a_{n-1} s + a_n) Y(s) = (b_0 s^m + \cdots + b_m) U(s) \tag{2-3}$$

输出拉普拉斯变换 $Y(s)$ 与输入拉普拉斯变换 $U(s)$ 之比

$$G(s) = \frac{Y(s)}{U(s)} = \frac{b_0 s^m + \cdots + b_{m-1} s + b_m}{a_0 s^n + \cdots + a_{n-1} s + a_n} \tag{2-4}$$

即为单输入-单输出系统的传递函数，其模型参数可表示为

传递函数分母系数向量 $\boldsymbol{A}=[a_0, a_1, \cdots, a_n]$，$n+1$ 维

传递函数分子系数向量 $\boldsymbol{B}=[b_0, b_1, \cdots, b_m]$，$m+1$ 维

分母多项式阶次 n

分子多项式阶次 m

用 $num = \boldsymbol{B}$，$den = \boldsymbol{A}$ 分别表示分子、分母参数向量，则可简练地表示为（num，den）形式，称为传递函数二对组模型参数。

式（2-4）中，当 $a_0 = 1$ 时，分子多项式成为

$$s^n + a_1 s^{n-1} + \cdots + a_{n-1} s + a_n \tag{2-5}$$

称为系统的首一特征多项式，是控制系统常用的标准表达形式，于是相应的模型参数中，分母系数向量只用 n 维分量即可表示出，即

$$\boldsymbol{A} = [a_1, a_2, \cdots, a_n], n \text{ 维}$$

（四）零极点增益形式

如果将式（2-4）中分子、分母有理多项式分解为因式连乘形式，则有

$$G(s) = K \frac{\prod_{i=1}^{m} (s - z_i)}{\prod_{j=1}^{n} (s - p_j)} = K \frac{(s - z_1)(s - z_2)\cdots(s - z_m)}{(s - p_1)(s - p_2)\cdots(s - p_n)} \tag{2-6}$$

式中，K 为系统的零极点增益；z_i，$i = 1, 2, \cdots, m$，称为系统的零点；p_j，$j = 1, 2, \cdots, n$，称为系统的极点（z_i、p_j 可以是实数，也可以是复数）。

称式（2-6）为单输入-单输出系统传递函数的零极点表达形式。其模型参数为

系统零点向量：$\boldsymbol{Z} = [z_1, z_2, \cdots, z_m]$，$m$ 维

系统极点向量：$\boldsymbol{P} = [p_1, p_2, \cdots, p_n]$，$n$ 维

系统零极点增益：K，标量

简记为（$\boldsymbol{Z}$，$\boldsymbol{P}$，K）形式，称为零极点增益三对组模型参数。

（五）部分分式形式

传递函数也可表示成为部分分式或留数形式，如下所示

$$G(s) = \sum_{i=1}^{n} \frac{r_i}{s - p_i} + h(s) \tag{2-7}$$

式中，p_i（$i = 1, 2, \cdots, n$）为该系统的 n 个极点，与零极点形式的 n 个极点是一致的；r_i（$i = 1, 2, \cdots, n$）是对应各极点的留数；$h(s)$则表示传递函数分子多项式除以分母多项式的余式，若分子多项式阶次与分母多项式相等，h 为标量，若分子多项式阶次小于分母多项式阶次，该项不存在。

模型参数表示为

极点留数向量：$\boldsymbol{R} = [r_1, r_2, \cdots, r_n]$，$n$ 维

系统极点向量：$\boldsymbol{P} = [p_1, p_2, \cdots, p_n]$，$n$ 维

余式系数向量：$\boldsymbol{H} = [h_0, h_1, \cdots, h_l]$，$l+1$ 维，且 $l = m - n$，原函数中分子大于分母阶次的余式系数。$l < 0$ 时，该向量不存在

简记为（$\boldsymbol{R}$，$\boldsymbol{P}$，$\boldsymbol{H}$）形式，称为极点留数模型参数。

二、数学模型的转换

以上所述的几种数学模型可以相互转换，以适应不同的仿真分析要求。

（一）微分方程与传递函数形式

微分方程的模型参数向量与传递函数的模型参数向量完全一样，所以微分方程模型在仿真中总是用其对应的传递函数模型来描述。

（二）传递函数与零极点增益形式

传递函数转化为零极点增益表示形式的关键，实际上取决于如何求取传递函数分子、分母多项式的根。令

$$b_0 s^m + b_1 s^{m-1} + \cdots + b_{m-1} s + b_m = 0 \tag{2-8}$$

$$a_0 s^n + a_1 s^{n-1} + \cdots + a_{n-1} s + a_n = 0 \tag{2-9}$$

则两式分别有 m 个和 n 个相应的根 z_i（$i=1, 2, \cdots, m$）和 p_j（$j=1, 2, \cdots, n$），此即为系统的 m 个零点和 n 个极点。求根过程可通过高级语言编程实现，但编程较繁琐。直接采用功能强大的 MATLAB 语言，可使模型转换过程变得十分方便。

MATLAB 语言的控制系统工具箱中提供了大量的实用函数，关于模型转换函数有好几种，其中 tf2zp（ ）和 zp2tf（ ），就是用来进行传递函数形式与零极点增益形式之间的相互转换的。

如语句：[**Z**, **P**, *K*] = tf2zp (*num*, *den*)

表示将分子、分母多项式系数向量为 *num*，*den* 的传递函数模型参数经运算返回左端式中的相应变量单元，形成零、极点表示形式的模型参数向量 **Z**、**P**、*K*。

同理，语句：[*num*, *den*] = zp2tf (**Z**, **P**, *K*)

也可方便地将零、极点增益形式表示为传递函数有理多项式形式。

（三）状态方程与传递函数或零极点增益形式

对于单变量系统，状态方程为

$$\begin{cases} \dot{\boldsymbol{X}} = \boldsymbol{AX} + \boldsymbol{BU} \\ \boldsymbol{Y} = \boldsymbol{CX} + \boldsymbol{DU} \end{cases} \tag{2-10}$$

可得

$$G(s) = \frac{Y(s)}{U(s)} = \boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B} + \boldsymbol{D} \tag{2-11}$$

关键在于 $(s\boldsymbol{I} - \boldsymbol{A})^{-1}$ 的求取。

利用 Fadeev-Fadeeva 法可以由已知的 **A** 阵求得 $(s\boldsymbol{I} - \boldsymbol{A})^{-1}$，并采用计算机高级语言（如 C 或 FORTRAN 等）编程实现。同样，通过使用 MATLAB 语言控制系统工具箱中提供的有关状态方程与传递函数的相互转换函数，ss2tf（ ）和 tf2ss（ ），可使转换过程大为简化。

如语句：[*num*, *den*] = ss2tf (**A**, **B**, **C**, **D**)

表示把描述为 (**A**, **B**, **C**, **D**) 的系统状态方程模型参数各矩阵转换为传递函数模型参数各向量。左式中的变量单元 *num* 即为转换函数返回的分子多项式参数向量；*den* 即为转换函数返回的分母多项式参数向量。于是

$$num = [b_0, b_1, \cdots, b_m]$$

$$den = [a_0, a_1, \cdots, a_n]$$

而语句 [**A**, **B**, **C**, **D**] = tf2ss (*num*, *den*) 是上述过程的逆过程，由已知的 (*num*, *den*) 经模型转换返回状态方程各参数矩阵 (**A**, **B**, **C**, **D**)。

需要说明的是，由于同一传递函数的状态方程实现不唯一，故上面所述的转换函数只能实现可控标准型状态方程。

转换函数 ss2zp（ ）和 zp2ss（ ）则是用以完成状态方程和零极点增益模型相互转换的功能函数。语句格式为

$$[\boldsymbol{Z},\boldsymbol{P},K] = \text{ss2zp}(\boldsymbol{A},\boldsymbol{B},\boldsymbol{C},\boldsymbol{D})$$

$$[\boldsymbol{A},\boldsymbol{B},\boldsymbol{C},\boldsymbol{D}] = \text{zp2ss}(\boldsymbol{Z},\boldsymbol{P},K)$$

（四）部分分式与传递函数或零极点增益形式

传递函数转化为部分分式的表示形式，关键在于求取各分式的分子待定系数，即下式中的 r_i（$i=1, 2, \cdots, n$）

$$G(s) = \frac{r_1}{s-p_1} + \frac{r_2}{s-p_2} + \cdots + \frac{r_n}{s-p_n} + h(s) \tag{2-12}$$

单极点情况下，该待定系数可用以下极点留数的求取公式得到

$$r_i = G(s)(s-p_i)\big|_{s=p_i} \tag{2-13}$$

具有多重极点时，也有相应极点留数的求取公式可选用，此处不作详细讨论。但无论如何，这些公式的应用或是根据公式算法编制程序的过程都相当麻烦。

MATLAB 语言中有专门解决极点留数求取的功能函数 residue（ ），可以非常方便地得到我们所需的结果。

语句 $[R, P, H]$ = residue (num, den)

$[num, den]$ = residue (R, P, H)

就是用来将传递函数形式与部分分式形式的数学模型相互转换的函数。

由上可知，数学模型可根据仿真分析需要建立为不同的形式，并利用 MATLAB 语言能够非常容易地相互转换，以适应仿真过程中的一些特殊要求。

三、线性时不变系统的对象数据类型描述

在新版的 MATLAB 语言（MATLAB5. x 以上版本）中，增添了“对象数据类型”，相应地，在控制系统工具箱中也定义了一些线性时不变（Linear Time Invariant）模型对象，即 LTI 对象。这种对象数据类型的引入，使得控制系统各种数学模型的描述和相互转换更为方便和简洁。

已知一个系统的模型参数，采用 LTI 对象数据形式建立系统模型有以下几种语句函数：

$\boldsymbol{G}$ = tf（num, den）　　利用传递函数二对组生成 LTI 对象模型

$\boldsymbol{G}$ = zpk（$\boldsymbol{Z}$, $\boldsymbol{P}$, $\boldsymbol{K}$）　　利用零极点增益三对组生成 LTI 对象模型

$\boldsymbol{G}$ = ss（$\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, $\boldsymbol{D}$）　　利用状态方程四对组生成 LTI 对象模型

LTI 对象模型 $\boldsymbol{G}$ 一旦生成，就可以用单一的变量名 $\boldsymbol{G}$ 描述系统的数学模型，非常便于将系统模型作为一个整体进行各种形式的转换和处理，而不必每次调用系统模型都需输入模型参数组各向量或矩阵数据。

调用以下功能函数语句，可方便地实现 LTI 对象数据类型下不同数学模型的转换：

$\boldsymbol{G}1$ = tf（$\boldsymbol{G}$）　　将 LTI 对象转换为传递函数模型

$\boldsymbol{G}2$ = zpk（$\boldsymbol{G}$）　　将 LTI 对象转换为零极点增益模型

$\boldsymbol{G}3$ = ss（$\boldsymbol{G}$）　　将 LTI 对象转换为状态方程模型

也可通过调用以下函数获得不同要求下的模型参数组向量或矩阵数据：

[*num*，*den*] = tfdata（*G*）　　从 LTI 对象获取传递函数二对组模型参数

[*Z*，*P*，*K*] = zpkdata（*G*）　　从 LTI 对象获取零极点增益三对组模型参数

[*A*，*B*，*C*，*D*] = ssdata（*G*）　从 LTI 对象获取状态方程四对组模型参数

利用“LTI 对象模型”可直接进行各种系统分析，控制系统工具箱中许多功能函数均能照常调用，使得对系统的仿真、分析效率大大提高。有关对象数据类型更为深入的应用和详细内容、方法等请参阅相关文献资料，受本书篇幅所限，这里只能对与后续章节相关的内容予以简介。

四、控制系统建模的基本方法

控制系统数学模型的建立是否得当，将直接影响以此为依据的仿真分析与设计的准确性、可靠性，因此必须予以充分重视，以采用合理的方式方法。

（一）机理模型法

所谓机理模型，实际上就是采用由一般到特殊的推理演绎方法，对已知结构、参数的物理系统运用相应的物理定律或定理，经过合理分析简化而建立起来的描述系统各物理量动、静态变化性能的数学模型。

因此，机理模型法主要是通过理论分析推导方法建立系统模型。根据确定元件或系统行为所遵循的自然机理，如常用的物质不灭定律（用于液位、压力调节等）、能量守恒定律（用于温度调节等）、牛顿第二定律（用于速度、加速度调节等）、基尔霍夫定律（用于电气网络）等，对系统各种运动规律的本质进行描述，包括质量、能量的变换和传递等过程，从而建立起变量间相互制约又相互依存的精确的数学关系。通常情况下，是给出微分方程形式或其派生形式——状态方程、传递函数等。

建模过程中，必须对控制系统进行深入地分析研究，善于提取本质、主流方面的因素，忽略一些非本质、次要的因素，合理确定对系统模型准确度有决定性影响的物理变量及其相互作用关系，适当舍弃对系统性能影响微弱的物理变量和相互作用关系，避免出现冗长、复杂、繁琐的公式方程堆砌。最终目的是要建造出既简单清晰，又具有相当精度，能够反映实际物理量变化的控制系统数学模型。

建立机理模型还应注意所研究系统模型的线性化问题。大多数情况下，实际控制系统由于种种因素的影响，都存在非线性现象，如机械传动中的死区间隙、电气系统中磁路饱和等，严格地说都属于非线性系统，只是其非线性程度有所不同。在一定条件下，可以通过合理的简化、近似，用线性系统模型近似描述非线性系统。其优点在于可利用线性系统许多成熟的计算分析方法和特性，使控制系统的分析、设计更为简单方便，易于实用。但也应指出，线性化处理方法并非对所有控制系统都适用，对于包含本质非线性环节的系统需要采用特殊的研究方法。

（二）统计模型法

所谓统计模型法，就是采用由特殊到一般的逻辑、归纳方法，根据一定数量的在系统运行过程中实测、观察的物理量数据，运用统计规律、系统辨识等理论合理估计出反映系统各物理量相互制约关系的数学模型。其主要依据是来自系统的大量实测数据，因此又称之为实验测定法。

当对所研究系统的内部结构和特性尚不清楚、甚至无法了解时，系统内部的机理变化规律就不能确定，通常称之为“黑箱”或“灰箱”问题，机理模型法也就无法应用。而根据所测到的系统输入、输出数据，采用一定方法进行分析及处理来获得数学模型的统计模型法正好适应这种情况。通过对系统施加激励，观察和测取其响应，了解其内部变量的特性，并建立能近似反映同样变化的模拟系统的数学模型，就相当于建立起实际系统的数学描述（方程、曲线或图表等）。

频率特性法是研究控制系统的一种应用广泛的工程实用方法。其特点在于通过建立系统频率响应与正弦输入信号之间的稳态特性关系，不仅可以反映系统的稳态性能，而且可以用来研究系统的稳定性和暂态性能；可以根据系统的开环频率特性，判别系统闭环后的各种性能；可以较方便地分析系统参数对动态性能的影响，并能大致指出改善系统性能的途径。

频率特性物理意义十分明确，对稳定的系统或元件、部件都可以用实验方法确定其频率特性，尤其对一些难以列写动态方程、建立机理模型的系统，有特别重要的意义。

系统辨识法是现代控制理论中常用的技术方法，它也是依据观察到的输入与输出数据来估价动态系统的数学模型的，但输出响应不局限于频率响应，阶跃响应或脉冲响应等时间响应都可作为反映系统模型动态特性的重要信息，且确定模型的过程更依赖于各种高效率的最优算法以及如何保证所测取数据的可靠性等理论问题。因其在实践中能得到很好的运用，故已被广泛接受，并逐渐发展成为较成熟且日臻完善的一门学科。

应当注意，由于对系统了解得不很清楚，主要靠实验测取数据确定数学模型的方法受数据量不充分、数据精度不一致、数据处理方法不完善等局限性影响，所得的数学模型的准确度只能满足一般工程需要，难以达到更高精度的要求。

（三）混合模型法

除以上两种方法外，控制系统还有这样一类问题，即对其内部结构和特性有部分了解，但又难以完全用机理模型方法表述出来，这时需结合一定的实验方法确定另外一部分不甚了解的结构特性，或是通过实际测定来求取模型参数。这种方法是机理模型法和统计模型法的结合，故称混合模型法。实用中它可能比前两者都用得多，是一项很好的理论推导与实验分析相结合的方法与手段。

控制系统的建模是一个理论性与实践性都很强的问题，是影响数字仿真结果的首要因素，鉴于本书的篇幅有限，此处不再展开讨论。下面的例题有助于对这一问题的理解。

例 2-1 控制系统原理图如图 2-1 所示，运用机理模型法建立系统的数学模型。

解 由系统原理图可知系统为一位置伺服闭环控制系统，将其分解为基本元件或部件，按工作机理分别列写输入-输出动态方程，并按各元件、部件之间的关系，画出系统结构图，最后根据结构图求出系统的总传递函数，从而建立起系统的数学模型。

1）同步误差检测器。设其输入为给定角位移 θ_r 与实际角位移 θ_c 之差，输出为位移误差电压 u_1，且位移/电压转换系数为 k_1，所以有

$$u_1 = k_1(\theta_r - \theta_c)$$

2）放大器。设其输入为位移误差电压 u_1 与测速发电机反馈电压 u_2 之差，输出为直流电动机端电压 u，电压放大系数 k_2，则有

$$u = k_2(u_1 - u_2)$$

3）直流电动机。设其输入为 u，输出为电动机角速度 ω，R 为电枢回路电阻，L 为电枢

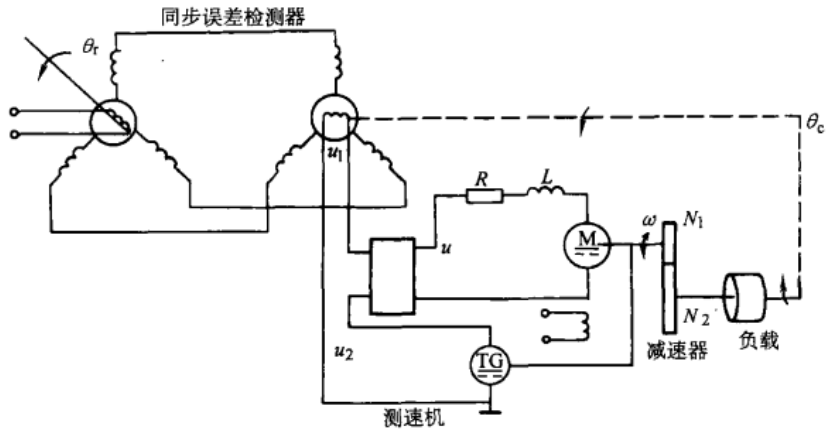


图 2-1　例 2-1 控制系统的原理图

回路电感，k_m 为电磁转矩系数，J 为电动机转动惯量，忽略反电动势和负载转矩影响，则由电动机电压平衡方程和力矩平衡方程，有

$$u = L\frac{\mathrm{d}i_a}{\mathrm{d}t} + Ri_a$$

$$k_m i_a = J\frac{\mathrm{d}\omega}{\mathrm{d}t}$$

所以

$$T\frac{\mathrm{d}^2\omega}{\mathrm{d}t^2} + \frac{\mathrm{d}\omega}{\mathrm{d}t} = k_3 u$$

式中，T 为电动机电磁时间常数，$T = \dfrac{L}{R}$；k_3 为电压/速度转换系数，$k_3 = \dfrac{k_m}{RJ}$。

推导中消去了中间变量——电枢电流 i_a。

4）测速发电机。设其输入为电动机角速度 ω，输出为测速电压值 u_2，速度/电压转换系数为 k_4，所以有

$$u_2 = k_4\omega$$

5）负载输出。设输入为电动机角速度 ω，输出为负载角位移 θ_c，传动比 $n = N_1/N_2 < 1$，则

$$\frac{\mathrm{d}\theta_c}{\mathrm{d}t} = n\omega$$

$\dfrac{U_1(s)}{\theta_r(s) - \theta_c(s)} = k_1$　　$\dfrac{U(s)}{U_1(s) - U_2(s)} = k_2$

$\dfrac{\omega(s)}{U(s)} = \dfrac{k_3}{s(Ts+1)}$　　$\dfrac{u_2(s)}{\omega(s)} = k_4$　　$\dfrac{\theta_c(s)}{\omega(s)} = \dfrac{n}{s}$

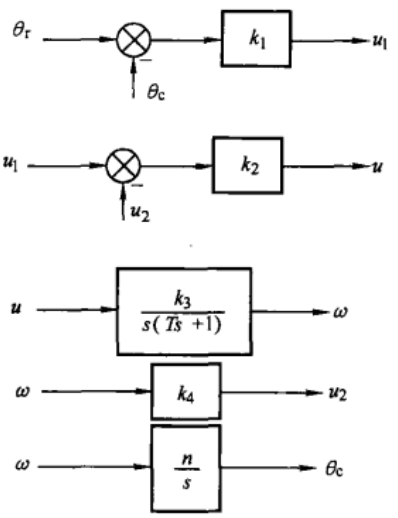


图 2-2　例 2-1 各环节传递函数及其结构图

6）将 1）～5）中各式进行拉普拉斯变换，注意变换后各变量像函数均为大写形式，按输入/输出关系表示出各环节传递函数，并据此画出各部分的结构图如图 2-2 所示。

7）按相互之间作用关系，连成系统总结构图，如图 2-3 所示。然后利用结构图等效变换化简或直接运用梅逊公式，求出该系统总传递函数 $G_B(s)$，得

$$G_B(s) = \frac{\theta_c(s)}{\theta_r(s)} = \frac{k_1k_2k_3n}{Ts^3 + s^2 + k_2k_3k_4s + k_1k_2k_3n}$$

即为所需的系统数学模型。

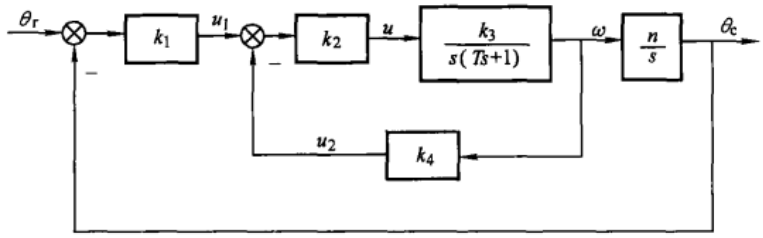


图 2-3 例 2-1 控制系统总结构图

例 2-2 用实验方法测得某系统的开环频率响应数据见表 2-1。试由表中数据建立该系统开环传递函数模型 $G(s)$。

表 2-1 例 2-2 系统的开环频率响应实测数据表

$\omega/(\text{rad}\cdot\text{s}^{-1})$	0.10	0.14	0.23	0.37	0.60	0.95	1.53	2.44	3.91	6.25	10.0
$L(\omega)/\text{dB}$	-0.049	-0.102	-0.258	-0.638	-1.507	-3.270	-6.315	-10.81	-16.69	-23.65	-31.27
$\phi(\omega)/(°)$	-9.72	-14.12	-22.45	-35.35	-54.56	-81.25	-115.5	-157.2	-207.8	-271.7	-358.9

注：ω 为输入信号角频率；$L(\omega)$ 为输出信号对数幅频特性值；$\phi(\omega)$ 为输出信号对数相频特性值。

解 1）由已知数据绘制该系统开环频率响应伯德图，如图 2-4 所示。

2）用 $\pm 20dB/dec$ 及其倍数的折线逼近幅频特性，如图 2-4 中折线，得两个转折频率，即

$$\omega_1 = 1\text{rad/s}, \quad \omega_2 = 2.85\text{rad/s}$$

求出相应惯性环节的时间常数为

$$T_1 = \frac{1}{\omega_1} = 1\text{s}, \quad T_2 = \frac{1}{\omega_2} = 0.35\text{s}$$

3）由低频段幅频特性可知，$L(\omega)|_{\omega\to 0} = 0$，所以 $K = 1$。

4）由高频段相频特性可知，相位滞后已超过 $-180°$，且随 ω 增大，滞后愈加严重，显然该系统存在纯滞后环节 $e^{-\tau s}$，为非最小相位系统。因此，系统开环传递函数应为以下形式

$$G(s) = \frac{Ke^{-\tau s}}{(T_1s+1)(T_2s+1)} = \frac{1}{(s+1)(0.35s+1)}e^{-\tau s}$$

5）设法确定纯滞后时间 τ 值。查图中 $\omega = \omega_1 = 1\text{rad/s}$ 时，$\phi(\omega_1) = -86°$，而按所求得的传递函数，应有

$$\phi(\omega_1) = -\arctan 1 - \arctan 0.35 - \tau_1 \times \frac{180°}{\pi} = -86°$$

易解得

$$\tau_1 = 0.37\text{s}$$

再查图中 $\omega = \omega_2 = 2.85\text{rad/s}$ 时，$\phi(\omega_2) = -169°$，同样从

$$\phi(\omega_2) = -\arctan 2.85 - \arctan(0.35 \times 2.85) - 2.85\tau_2 \times \frac{180°}{\pi}$$

$$= -169°$$

解得　　$\tau_2 = 0.33\text{s}$

取两次平均值得　　$\tau = \dfrac{\tau_1 + \tau_2}{2} = 0.35\text{s}$

6）最终求得该系统开环传递函数模型 $G(s)$ 为

$$G(s) = \frac{Ke^{-\tau s}}{(T_1 s + 1)(T_2 s + 1)} = \frac{1}{(s + 1)(0.35s + 1)} \times e^{-0.35s}$$

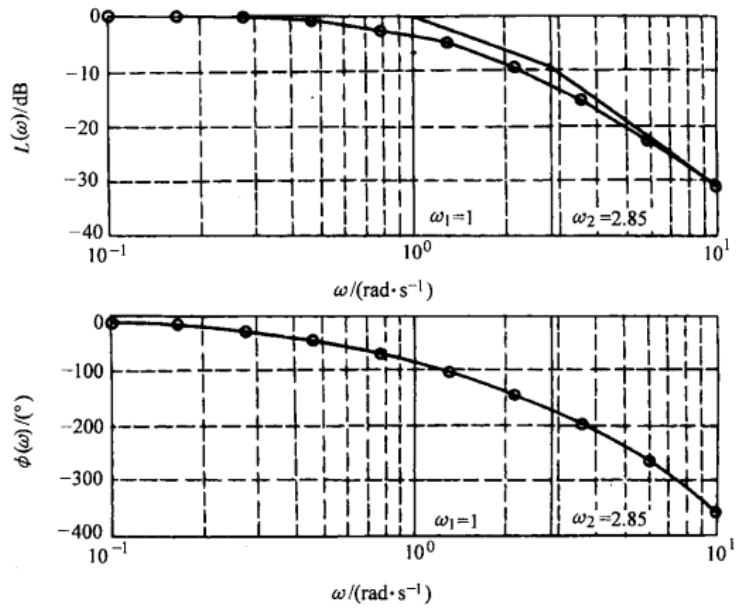


图 2-4　例 2-2 系统的开环频率响应伯德图

从以上两例可体会到，无论采用何种方法建模，其实质就是设法获取关于系统尽可能多的信息并经过恰当信息处理而得到对系统准确合理的描述。物理定律公式、实测试验数据等都是反映系统性能的重要信息，机理模型法、统计模型法只是信息处理过程不同而已，在实际建模过程中应灵活掌握运用。

第二节　控制系统建模实例

一、独轮自行车实物仿真问题

（一）问题提出

为实现图 2-5 所示的娱乐型“独轮自行车机器人”，控制工程师研制了图 2-6 所示的实物仿真模型。通过对该实物模型的理论分析与实物仿真实验研究，有助于实现对独轮自行车机器人的有效控制。

控制理论中把此问题归结为“一阶直线倒立摆控制问题”，如图 2-7 所示。另外，诸如机器人行走过程中的平衡控制、火箭发射中的垂直度控制、卫星飞行中的姿态控制、海上钻井平台的稳定控制、飞机安全着陆控制等均涉及到倒立摆的控制问题。

（二）建模机理

由于此问题为“单一刚性铰链、两自由度动力学问题”，因此，依据经典力学的牛顿定律即可满足要求。

（三）系统建模

如图 2-7 所示，设小车的质量为 m_0，倒立摆均匀杆的质量为 m，摆长为 $2l$，摆的偏角为 θ，小车的位移为 x，作用在小车上的水平方向的力为 F，O_1 为摆杆的质心。

根据刚体绕定轴转动的动力学微分方程，转动惯量与角加速度乘积等于作用于刚体主动力对该轴力矩的代数和，则

1）摆杆绕其重心的转动方程为

$$J\ddot{\theta} = F_y l\sin\theta - F_x l\cos\theta \tag{2-14}$$



图 2-5 独轮自行车

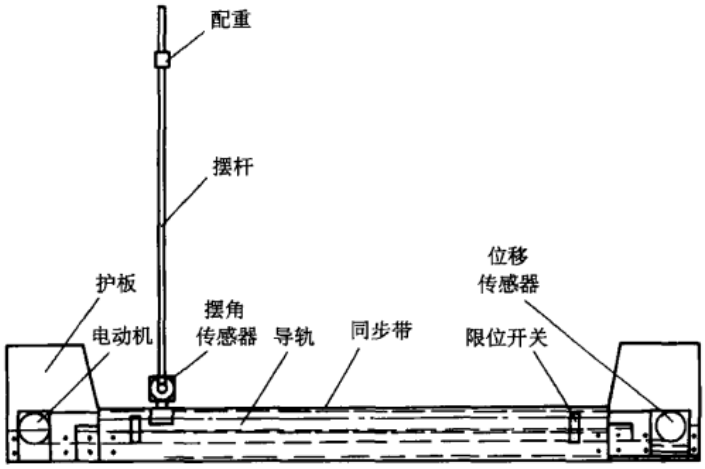


图 2-6 独轮自行车实物仿真模型

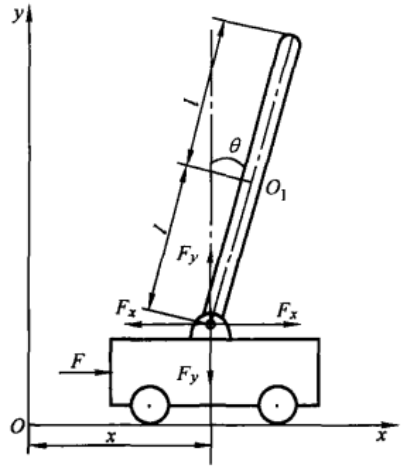


图 2-7 一阶倒立摆的物理模型

2）摆杆重心的水平运动可描述为

$$F_x = m\frac{d^2}{dt^2}(x + l\sin\theta) \tag{2-15}$$

3）摆杆重心在垂直方向上的运动可描述为

$$F_y - mg = m\frac{d^2}{dt^2}(l\cos\theta) \tag{2-16}$$

4）小车水平方向运动可描述为

$$F - F_x = m_0\frac{d^2x}{dt^2} \tag{2-17}$$

由式（2-15）和式（2-17）得

$$(m_0 + m)\ \ddot{x} + ml\ (\cos\theta \cdot \ddot{\theta} - \sin\theta \cdot \dot{\theta}^2)\ = F \tag{2-18}$$

由式（2-14）、式（2-15）和式（2-16）得

$$(J + ml^2)\ \ddot{\theta} + ml\cos\theta \cdot \ddot{x} = mlg\sin\theta \tag{2-19}$$

整理式（2-18）和式（2-19），得

$$\begin{cases} \ddot{x} = \dfrac{(J+ml^2)F + lm(J+ml^2)\sin\theta \cdot \dot{\theta}^2 - m^2l^2g\sin\theta\cos\theta}{(J+ml^2)(m_0+m) - m^2l^2\cos^2\theta} \\ \ddot{\theta} = \dfrac{ml\cos\theta \cdot F + m^2l^2\sin\theta\cos\theta \cdot \dot{\theta}^2 - (m_0+m)mlg\sin\theta}{m^2l^2\cos^2\theta - (m_0+m)(J+ml^2)} \end{cases} \tag{2-20}$$

因为摆杆是均质细杆，所以可求其对于质心的转动惯量。因此设细杆摆长为 $2l$，单位长度的质量为 ρ_l，取杆上一个微段 $\mathrm{d}x$，其质量为 $m=\rho_l \mathrm{d}x$，则此杆对于质心的转动惯量有

$$J = \int_{-l}^{l} (\rho_l \mathrm{d}x)x^2 = 2\rho_l l^3/3$$

杆的质量为

$$m = 2\rho_l l$$

所以此杆对于质心的转动惯量有

$$J = \frac{ml^2}{3}$$

（四）模型简化

由式（2-20）可见，一阶直线倒立摆系统的动力学模型为非线性微分方程组。为了便于应用经典控制理论对该控制系统进行设计，必须将其简化为线性定常的系统模型。

若只考虑 θ 在其工作点 $\theta_0=0$ 附近（$-10° < \theta < 10°$）的细微变化，则可近似认为

$$\begin{cases} \dot{\theta}^2 \approx 0 \\ \sin\theta \approx \theta \\ \cos\theta \approx 1 \end{cases}$$

在这一简化思想下，系统精确模型式(2-20)可简化为

$$\begin{cases} \ddot{x} = \dfrac{(J+ml^2)F - m^2l^2g\theta}{J(m_0+m) + mm_0l^2} \\ \ddot{\theta} = \dfrac{(m_0+m)mlg\theta - mlF}{J(m_0+m) + mm_0l^2} \end{cases}$$

若给定一阶直线倒立摆系统的参数为：小车的质量 $m_0=1\mathrm{kg}$；倒摆振子的质量 $m=1\mathrm{kg}$；倒摆长度 $2l=0.6\mathrm{m}$；重力加速度取 $g=10\mathrm{m/s^2}$，则可得到进一步简化模型为

$$\begin{cases} \ddot{x} = -6\theta + 0.8F \\ \ddot{\theta} = 40\theta - 2.0F \end{cases} \tag{2-21}$$

上式为系统的“微分方程模型”，对其进行拉普拉斯变换可得系统的传递函数模型为

$$\begin{cases} G_1(s) = \dfrac{\theta(s)}{F(s)} = \dfrac{-2.0}{s^2-40} \\ G_2(s) = \dfrac{X(s)}{\theta(s)} = \dfrac{-0.4s^2+10}{s^2} \end{cases} \tag{2-22}$$

图 2-8 为系统的动态结构图。

同理可得系统的状态方程模型。设系统状态为

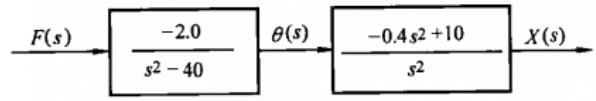


图 2-8 系统动态结构图

$$\boldsymbol{X}=\begin{bmatrix}x_1\\x_2\\x_3\\x_4\end{bmatrix}=\begin{bmatrix}\theta\\\dot{\theta}\\x\\\dot{x}\end{bmatrix}$$

则有系统状态方程

$$\dot{\boldsymbol{X}}=\begin{bmatrix}\dot{x}_1\\\dot{x}_2\\\dot{x}_3\\\dot{x}_4\end{bmatrix}=\begin{bmatrix}0&1&0&0\\40&0&0&0\\0&0&0&1\\-6&0&0&0\end{bmatrix}\begin{bmatrix}x_1\\x_2\\x_3\\x_4\end{bmatrix}+\begin{bmatrix}0\\-2\\0\\0.8\end{bmatrix}F=\boldsymbol{AX}+\boldsymbol{BF} \tag{2-23}$$

输出方程

$$\boldsymbol{Y}=\begin{bmatrix}\theta\\x\end{bmatrix}=\begin{bmatrix}1&0&0&0\\0&0&1&0\end{bmatrix}\begin{bmatrix}x_1\\x_2\\x_3\\x_4\end{bmatrix}=\boldsymbol{CX} \tag{2-24}$$

由此可见，通过对系统模型的简化，得到了一阶直线倒立摆系统的微分方程、传递函数、状态方程三种线性定常的数学模型，这为下面的系统设计奠定了基础。

（五）模型验证

对于所建立的一阶直线倒立摆系统数学模型，还应对其可靠性进行验证，以保证以后系统数字仿真实验的真实、有效。

一阶直线倒立摆系统的模型验证问题请读者参见本书第四章第三节中的相关内容。鉴于篇幅所限，有关模型验证部分的理论与方法问题，感兴趣的读者请参见相关专著。

二、龙门起重机运动控制问题

（一）问题提出

起重机，又名吊车，作为一种运载工具，广泛地应用于现代工厂、安装工地和集装箱货场以及室内外仓库的装卸与运输作业。它在离地面很高的轨道上运行，具有占地面积小、省工省时的优点，是工厂、仓库、码头必不可少的装卸搬运工具。图 2-9 为起重机的实物照片。



图 2-9 龙门起重机

起重机利用绳索一类的柔性体代替刚体工作，以使得起重机的结构轻便、工作

效率高。但是采用柔性体吊运也带来一些负面影响，例如起重机负载——重物的摆动问题一直是困扰提高起重机装运效率的一个难题。

为研究起重机的防摆控制问题，需对实际问题进行简化、抽象。起重机的“搬运—行走—定位”过程可抽象为如图 2-10 所示的情况。

其中，小车的质量为 m_0，受到水平方向的外力 $F(t)$ 的作用，重物的质量为 m，绳索的长度为 l。对重物的快速吊运与定位问题可以抽象为：求小车在所受的外力 $F(t)$ 的作用下，使得小车能在最短的时间 t_n 内由 A 点运动到 B 点，且 $|\theta(t_n)| < \Delta$，Δ 为系统允许的最小摆角。

（二）建模机理[58]

该问题为多刚体、多自由度、多约束的质点系动力学问题。由于牛顿经典力学主要是解决自由质点的动力学问题，对于自由质点系的动力学问题，是把物体系拆开成若干分离体，按反作用定律附加以约束反力，而后列写动力学方程。显然，对于龙门起重机运动系统的动力学问题应用牛顿力学来分析势必过于复杂。

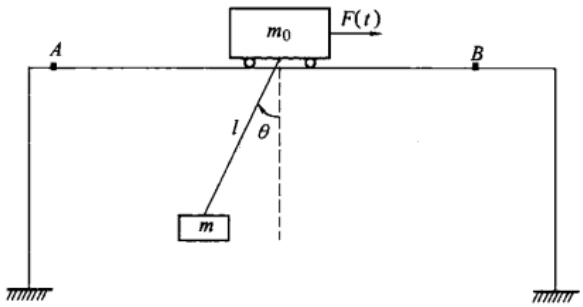


图 2-10　起重机系统的物理抽象模型

对于约束质点系统动力学问题，1788 年拉格朗日发表的名著《分析力学》一书中以“质点系统”为对象，应用虚位移与虚功原理，消除了系统中的约束力，得出了质点系平衡时主动力之间的关系。拉格朗日给出了解决具有完整约束的质点系动力学问题的具有普遍意义的方程，被后人称之为拉格朗日方程，它是分析力学中的重要方程。

拉格朗日方程的表达式非常简洁，应用时只需计算系统的动能和广义力。拉格朗日方程的普遍形式为

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial T}{\partial \dot{q}_k}\right)-\frac{\partial T}{\partial q_k}=F_k \qquad (k=1,2,\cdots,n) \tag{2-25}$$

式中，T 为质点系的动能，$T=\sum_{i=1}^{n}\frac{1}{2}m_i v_i^2$；$q_k$ 为质点系的广义坐标；n 为质点系的自由度；F_k 为广义力。

由此可见，拉格朗日方程把力学体系的运动方程从以力为基本概念的牛顿形式，改变为以能量为基本概念的分析力学形式。它奠定了分析力学的基础，为把力学理论推广应用到其他领域开辟了道路。

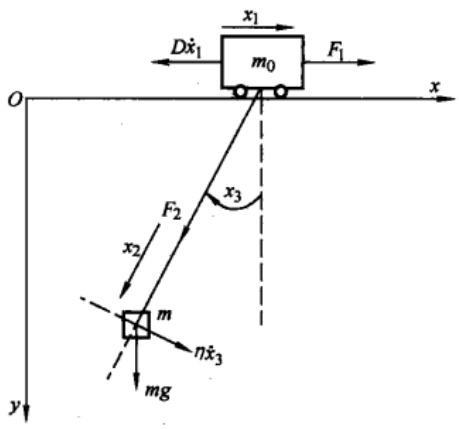


图 2-11　龙门起重机的物理模型

（三）系统建模

实际中的起重机系统受到多种干扰，如小车与导轨之间的干摩擦、风力的影响等，为了分析其本质，必须对实际系统进一步抽象。通过对龙门起重机进行分析，可将其抽象为如图 2-11 所示的物理模型。

重物通过绳索与小车相连，小车在行走电动机的水平拉力 F_1 的作用下在水平轨道上运

动，小车的质量为 m_0，重物的质量为 m，绳索的长度为 l，可在提升电动机的提升力 F_2 的作用之下进行升降运动；绳索的弹性、质量、运动的阻尼系数可忽略；小车与水平轨道的摩擦阻尼系数为 D；重物摆动时的阻尼系数为 η，其他扰动可忽略。

取小车位置为 x_1，绳长为 x_2，摆角为 x_3，作为系统的广义坐标系，在此基础上对系统进行动力学分析。

由图 2-11 所示的坐标系可知，小车和重物的位置坐标为

$$\begin{cases} x_{m_0} = x_1 \\ y_{m_0} = 0 \\ x_m = x_1 - x_2 \sin x_3 \\ y_m = x_2 \cos x_3 \end{cases} \tag{2-26}$$

所以小车和重物的速度分量为

$$\begin{cases} \dot{x}_{m_0} = \dot{x}_1 \\ \dot{y}_{m_0} = 0 \\ \dot{x}_m = \dot{x}_1 - \dot{x}_2 \sin x_3 - x_2 \dot{x}_3 \cos x_3 \\ \dot{y}_m = \dot{x}_2 \cos x_3 - x_2 \dot{x}_3 \sin x_3 \end{cases} \tag{2-27}$$

系统的动能为

$$\begin{aligned} T &= \frac{1}{2} m_0 v_{m_0}^2 + \frac{1}{2} m v_{m_0}^2 \\ &= \frac{1}{2} m_0 (\dot{x}_{m_0}^2 + \dot{y}_{m_0}^2) + \frac{1}{2} m (\dot{x}_m^2 + \dot{y}_m^2) \\ &= \frac{1}{2} (m_0 + m) \dot{x}_1^2 + \frac{1}{2} m (\dot{x}_2^2 + x_2^2 \dot{x}_3^2 - 2\dot{x}_1 \dot{x}_2 \sin x_3 - 2\dot{x}_1 x_2 \dot{x}_3 \cos x_3) \end{aligned} \tag{2-28}$$

此系统的拉格朗日方程组为

$$\begin{cases} \dfrac{\mathrm{d}}{\mathrm{d}t}\left(\dfrac{\partial T}{\partial \dot{x}_1}\right) - \dfrac{\partial T}{\partial x_1} = F_1 - D\dot{x}_1 \\ \dfrac{\mathrm{d}}{\mathrm{d}t}\left(\dfrac{\partial T}{\partial \dot{x}_2}\right) - \dfrac{\partial T}{\partial x_2} = F_2 + mg\cos x_3 \\ \dfrac{\mathrm{d}}{\mathrm{d}t}\left(\dfrac{\partial T}{\partial \dot{x}_3}\right) - \dfrac{\partial T}{\partial x_3} = -mgx_2 \sin x_3 - \eta \dot{x}_3 \end{cases} \tag{2-29}$$

综合以上公式得系统的方程组为

$$\begin{cases} (m + m_0)\ddot{x}_1 - m\ddot{x}_2 \sin x_3 - mx_2 \ddot{x}_3 \cos x_3 - 2m\dot{x}_2 \dot{x}_3 \cos x_3 + mx_2 \dot{x}_3^2 \sin x_3 + D\dot{x}_1 = F_1 \\ m\ddot{x}_2 - m\ddot{x}_1 \sin x_3 - mx_2 \dot{x}_3^2 - mg\cos x_3 = F_2 \\ mx_2^2 \ddot{x}_3 + 2mx_2 \dot{x}_2 \dot{x}_3 - m\ddot{x}_1 x_2 \cos x_3 + mgx_2 \sin x_3 + \eta \dot{x}_3 = 0 \end{cases} \tag{2-30}$$

式中，m 为重物的质量（kg）；m_0 为起重机的质量（kg）；g 为重力加速度（m/s^2）；D 为起重机的阻力系数（kg/s）；η 为重物运动的阻尼系数（$kg \cdot m^2/s$）；F_1 为小车受到水平方向

上的拉力（N）；F_2 为绳子受到提升电动机的拉力（N）。

式（2-30）即为考虑绳长变化情况下的二自由度起重机运动系统的动力学模型。

对于绳长保持不变的情况，可将上述模型进一步简化，将式（2-30）中的 $\dot{x}_2=\ddot{x}_2=0$，消去 F_2，令 $F=F_1$，$x_2=l=\text{const}$，可得到绳长不变时的起重机运动系统数学模型为

$$\begin{cases}(m_0+m)\ddot{x}_1+D\dot{x}_1-ml\ddot{x}_3\cos x_3+ml\dot{x}_3^2\sin x_3=F\\ ml^2\ddot{x}_3-m\ddot{x}_1l\cos x_3+mgl\sin x_3+\eta\dot{x}_3=0\end{cases}\tag{2-31}$$

（四）模型简化

由式（2-30）可见，龙门起重机运动系统的动力学模型为非线性微分方程组；为了便于应用经典控制理论对该控制系统进行设计，必须将其简化为线性定常的系统模型。

对于式（2-31）的定摆长起重机系统，其中 x/x_1 为小车的位置，θ/x_3 为重物摆角；F 是小车行走电动机的水平拉力，m_0 为小车的质量，m 为重物的质量，l 为绳索的长度，绳索运动的阻尼、弹性和质量可忽略；小车与水平轨道的摩擦阻尼系数为 D；重物摆动时的阻尼系数为 η，忽略其他扰动。

考虑到实际起重机运行过程中摆动角较小（不超过 10°），且平衡位置为 $\theta=0$，可将式（2-31）表示的模型在 $\theta=0$ 处进行线性化。此时有如下近似结果：$\sin\theta\approx\theta$，$\cos\theta\approx1$，$\dot{\theta}^2\sin\theta\approx0$。考虑到摆动的阻尼系数 η 较小，可认为 $\eta=0$，所以式（2-31）可简化为

$$\begin{cases}(m_0+m)\ddot{x}+D\dot{x}-ml\ddot{\theta}=F\\ ml\ddot{\theta}-m\ddot{x}+mg\theta=0\end{cases}\tag{2-32}$$

将式(2-32) 进一步化简得

$$\begin{cases}F=m_0\ddot{x}+D\dot{x}+mg\theta\\ \ddot{x}=l\ddot{\theta}+g\theta\end{cases}\tag{2-33}$$

对(2-33) 进行拉普拉斯变化可得

$$\begin{cases}F(s)=(m_0s^2+Ds)X(s)+mg\theta(s)\\ s^2X(s)=(ls^2+g)\theta(s)\end{cases}\tag{2-34}$$

由上面系统的传递函数形式，可得图 2-12 所示的定摆长起重机运动系统动态结构图，图 2-13 是其另一种表达形式。

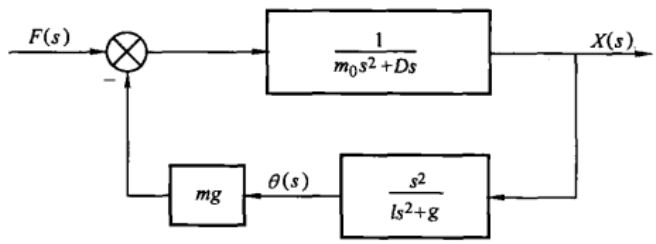


图 2-12　定摆长起重机运动系统动态结构图(形式一)

同理，也可将上述模型转化为状态空间形式。对式(2-33) 进行变换，每个式子只保留一个二次导数项，可得

$$\begin{cases} \ddot{x} = -\dfrac{D}{m_0}\dot{x} - \dfrac{mg}{m_0}\theta + \dfrac{1}{m_0}F \\ \ddot{\theta} = -\dfrac{D}{m_0 l}\dot{x} - \dfrac{(m_0+m)g}{m_0 l}\theta + \dfrac{1}{m_0 l}F \end{cases} \tag{2-35}$$

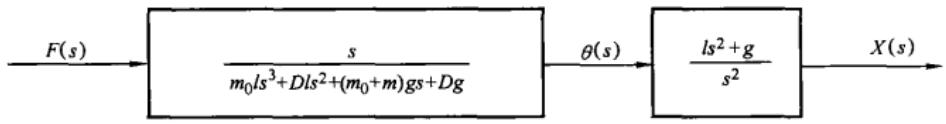


图 2-13 定摆长起重机运动系统动态结构图(形式二)

取 $x,\dot{x},\theta,\dot{\theta}$ 为系统的状态,x,θ 为系统的输出,则系统的状态空间描述方程为

$$\begin{cases} \dot{\boldsymbol{X}} = \boldsymbol{AX} + \boldsymbol{B}u \\ \boldsymbol{Y} = \boldsymbol{CX} \end{cases} \tag{2-36}$$

式中,$\boldsymbol{X} = [x,\dot{x},\theta,\dot{\theta}]^{\mathrm{T}}$,$u = F$,$\boldsymbol{Y} = [x,\theta]^{\mathrm{T}}$,

$$\boldsymbol{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\dfrac{D}{m_0} & -\dfrac{mg}{m_0} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -\dfrac{D}{m_0 l} & -\dfrac{(m_0+m)g}{m_0 l} & 0 \end{bmatrix},\quad \boldsymbol{B} = \begin{bmatrix} 0 \\ \dfrac{1}{m_0} \\ 0 \\ \dfrac{1}{m_0 l} \end{bmatrix},\quad \boldsymbol{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

式 (2-36) 即为定摆长起重机运动系统的状态空间表达式模型。

(五) 模型验证

模型验证参见本书第四章第五节的相关内容。

三、水箱液位控制问题

(一) 问题提出

图 2-14 所示为水箱液位控制原理图。在工业过程控制领域中，诸如电站锅炉汽包水位控制、化学反应釜液位控制、化工配料系统的液位控制等问题，均可等效为此水箱液位控制问题。图中，h 为液位高度（又称为稳态水头），q_{in}为流入水箱中液体的流量，q_{out}为流出水箱液体的流量，q'_{in}与 q'_{out}分别为进水阀门和出水阀门的控制开度，S 为水箱底面积。

(二) 建模机理

显然，此问题涉及到流体力学的理论，因此我们有必要就流体力学中的几个基本概念加以介绍。

(1) 雷诺系数 $Re = \dfrac{vd}{r}$，其中，Re 为雷诺系数，v 为液体流速，d 为管道口径，r 为液体粘度。

可见，雷诺系数反映了液体在管道中流动时的物理性能（流态）。

(2) 紊流 当流体的雷诺系数 $Re > 2000$ 时，流体的流态称为紊流。紊流流态表征了流体在传递中有能量损失，质点运动紊乱（有横向分量）。

1) 紊流条件下，流量 q（流速）与稳态水头 h（压力）有如下关系：$q = K\sqrt{h}$。

2）通常条件下，容器与导管连接处的流态呈紊流状态。

（3）层流　当流体的雷诺系数 $Re < 2000$ 时，流体的流态称为层流。

1）“层流”流态表征了流体在传递中能量损失很少，质点运动有序（沿轴向方向）。

2）层流条件下，流量 q（流速）与稳态水头 h（压力）的关系：$q = Kh$。

3）通常条件下，长距离直管段中，在压力恒定的情况下，流体呈层流状态。

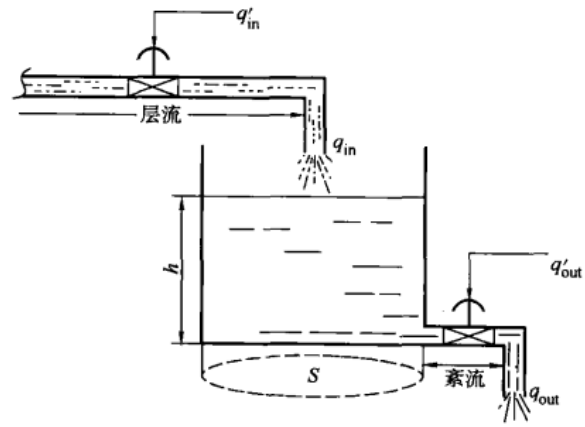


图 2-14　水箱液位控制原理图

（三）系统建模

由图 2-14 可知

$$h = K_3 \int (q_{in} - q_{out})\,dt$$

式中，$K_3 \propto S$（水箱底面积）。对上式取拉普拉斯变换得

$$H(s) = K_3 \frac{1}{s} [Q_{in}(s) - Q_{out}(s)]$$

综上有图 2-15a 所示的系统结构框图，将紊流与层流状态下 q 与 h 的关系代入其中，可得图 2-15b 所示的水箱液位系统动态结构图。

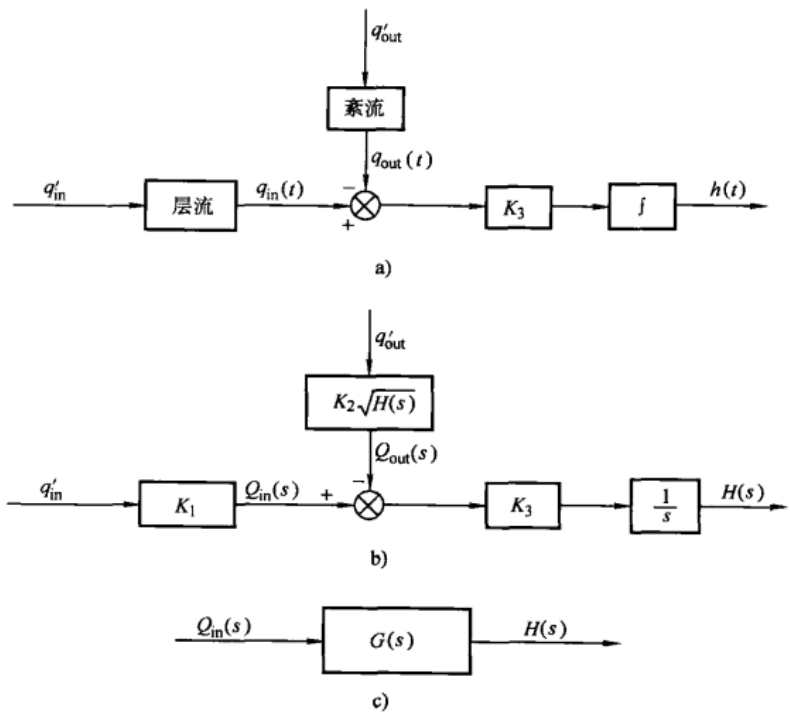


图 2-15　水箱液位控制系统框图

（四）模型简化

显然图 2-15b 所示的水箱液位系统为一非线性系统。为便于利用经典控制理论对该系统实施有效的设计，需将其在一定条件下简化为图 2-15c 所示的线性定常系统。

以下的模型简化系建立在“系统工作在平衡点附近”的条件之下，即系统中的 $q_{out}(t)$ 处于稳定状态。

（1）液阻与液容

定义 1 单位流量的变化所对应的液位差变化称为液阻，即

$$R=\frac{\mathrm{d}h}{\mathrm{d}q}=\frac{\text{液位差变化（单位为 m）}}{\text{流量变化（单位为 m}^3/\text{s）}}$$

定义 2 单位水头（液位）的变化所对应的被存储液体的变化称为液容，即

$$C=\frac{(q_{in}-q_{out})\,\mathrm{d}t}{\mathrm{d}h}=\frac{\text{被存储液体的变化（单位为 m}^3\text{）}}{\text{水头的变化（单位为 m）}}$$

显然，对于确定的水箱系统，液阻 R 与液容 C 是一个定数。

（2）平衡工作点 由于水箱系统的出水口处为紊流状态，即有

$$q=K\sqrt{h} \tag{2-37}$$

这样一个非线性关系存在，如图 2-16 所示。假设：水箱系统有一稳定的平衡工作点（q_0，h_0），则系统所在 P（q_0，h_0）处附近的（$\mathrm{d}q$，$\mathrm{d}h$）范围内可用直线代替曲线，该直线的斜率即为平衡工作点（q_0，h_0）处的液阻。

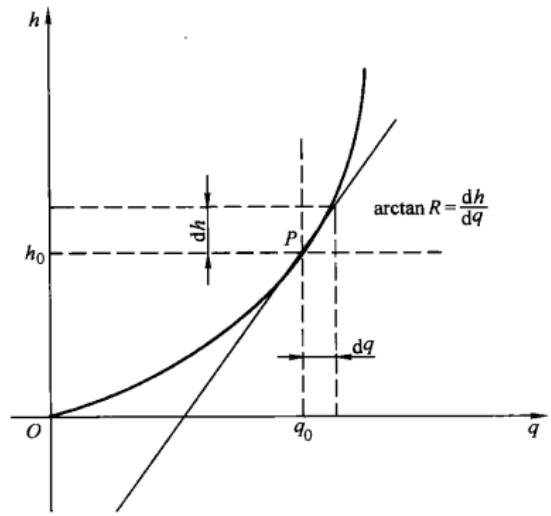


图 2-16 紊流状态下水头与流量的关系

由式（2-37）可知，$\mathrm{d}q=\frac{K}{2\sqrt{h}}\mathrm{d}h$，所以

$$\frac{\mathrm{d}h}{\mathrm{d}q}=2\sqrt{h}\frac{1}{K}=2\sqrt{h}\,\frac{\sqrt{h}}{q}=\frac{2h}{q}$$

因此，在水箱系统出水口处的液阻为 $R_0=\frac{2h}{q}$。

（3）模型简化 当水箱系统工作在系统平衡工作点附近时，可将图 2-15b 所示的非线性系统简化为一线性系统。

由液容定义知：$C\mathrm{d}h=(q_{in}-q_{out})\,\mathrm{d}t$

又由液阻定义知：$q_{out}=\frac{h}{R}$

则有 $$C\mathrm{d}h=\left(q_{in}-\frac{h}{R}\right)\mathrm{d}t$$

即 $$RC\frac{\mathrm{d}h}{\mathrm{d}t}+h=Rq_{in} \tag{2-38}$$

式中，RC 为水箱系统的时间常数。对式（2-38）取拉普拉斯变换（设初始条件为零）得

$$(RCs+1)H(s)=RQ_{in}(s)$$

则有 $$G(s)=\frac{H(s)}{Q_{in}(s)}=\frac{R}{RCs+1} \tag{2-39}$$

可见，在水箱系统平衡工作点附近，原非线性受扰系统可简化为无扰线性定常的惯性环节。

四、燃煤热水锅炉控制问题

（一）问题提出

图 2-17 所示为燃煤热水锅炉系统，在工业生产与民用集中供热等方面具有广泛的应用。图中，p（炉膛压力—微负压）、T_i（回水温度）、r（原煤燃值）、α（炉渣灰份）、T（烟气温度）与 w_{O_2}（废气含氧量）为锅炉燃烧质量监测量，T_o（出水温度）为燃烧热水锅炉的“被控量”。图 2-18 给出了上述系统的控制系统结构框图，从中可见，为利用数字仿真技术进行最佳燃烧控制器的设计，必须首先建立锅炉系统的数学模型，即 $\dfrac{T_o(s)}{\Phi_i(s)} = G(s)$，其中 Φ_i 为供给锅炉系统的有效热流量。

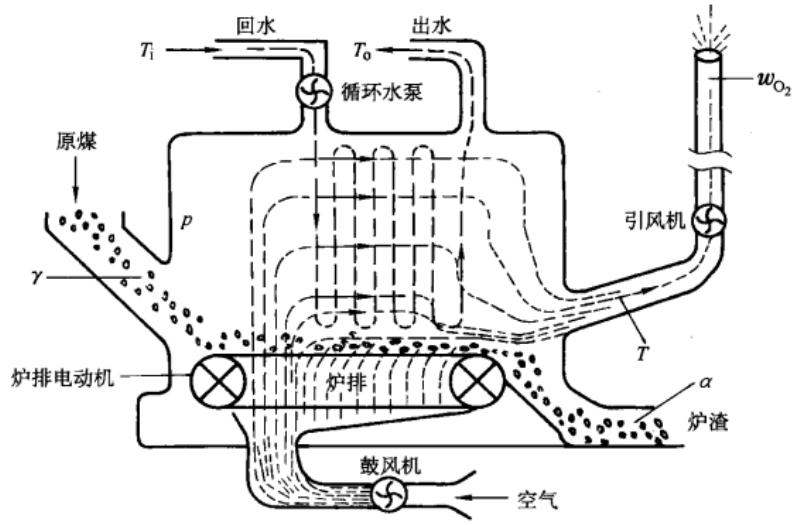


图 2-17　燃煤热水锅炉原理图

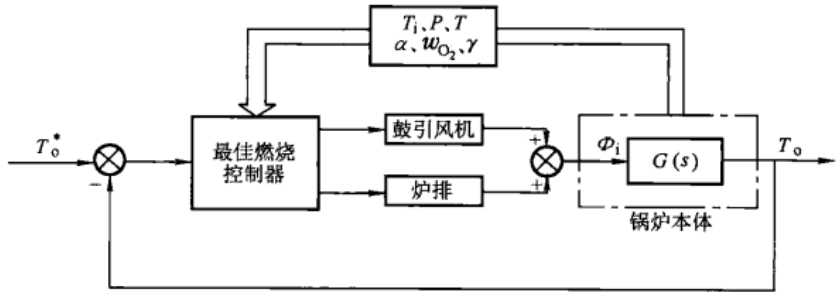


图 2-18　燃烧锅炉控制系统结构框图

（二）建模机理

显然，此问题涉及到热力学的理论，因此有必要就热力学中的几个概念加以介绍：

（1）热力学系统　将热量从一种物质传递到另一种物质的系统。

（2）热传递的三种途径　传导、对流、辐射。对于热传导有如下关系

$$\Phi = K\Delta T$$

式中，Φ 为热流量；K 为系数（反映了系统的导热性能）；ΔT 为系统温度的变化。

（3）热阻　用以描述热传导过程的“传导”性质（类似电阻），用下式表示

$$R = \frac{\Delta T}{\Delta \Phi} = \frac{1}{K} \tag{2-40}$$

对于热传导问题，R 是一个常量。

（4）热容　用以描述热力系统“保温”性质（类似电容），用下式表示

$$C = \frac{\Delta Q_{存储}}{\Delta T_{存储}} \tag{2-41}$$

式中，$\Delta Q_{存储}$为热力系统存储的热量的变化。对于理想的热力系统，C 通常是一个常数。

（三）系统建模

为便于建立系统的模型，上述燃煤热水锅炉可等效为图 2-19 所示的物理模型。

设系统保温良好（C 较大），炉膛内温度均匀（混合器的作用），则有

$$\Delta Q_{存储} = \Sigma \Phi_i - \Phi_o = \Phi_i + \Phi_{ei} - \Phi_o$$

式中，Φ_i 为由燃煤给出的热流量；Φ_{ei}为由回水给出的热流量；Φ_o为锅炉出口处的热流量。

由式（2-40）、式（2-41）可得

$$C_1 \Delta T_o = \Phi_i + \frac{T_i}{R_1} - \frac{T_o}{R_1}$$

化简得　$R_1 C_1 \Delta T_o + T_o = T_i + \Phi_i R_1$

取拉普拉斯变换得　$(R_1 C_1 s + 1) T_o(s) = T_i(s) + R_1 \Phi_i(s)$。

其等效的动态结构图如图 2-20 所示。

同理，不难推导出系统用户模型 $G_2(s) = \dfrac{1}{R_2 C_2 s + 1}$。

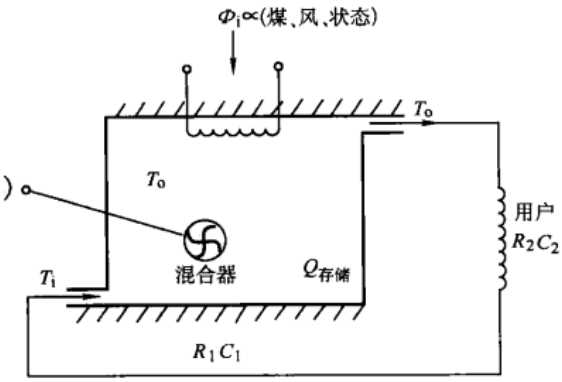


图 2-19　燃烧热水锅炉的物理模型

由上可知，只要确定 R_1、C_1（锅炉自身的特性）与 R_2、C_2（用户散热器特性），即可求解锅炉系统的热力学性能的数学模型。

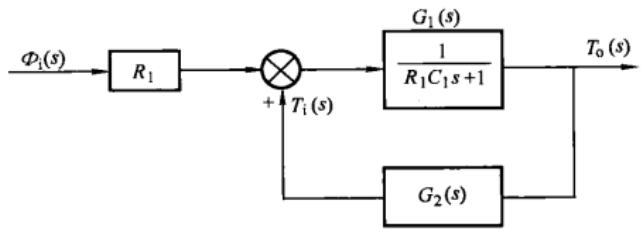


图 2-20　系统动态结构图

（四）存在问题

通过上述模型的建立过程，可以发现如下问题：

（1）分布参数问题　在实际的燃煤热水锅炉系统中，三种热传递形式同时存在；保温性能不佳，鼓引风存在泄漏等问题也不可避免的存在。因此，上述模型往往与实际问题存在一定的偏差。

（2）最佳燃烧控制问题　实际工作中，Φ_i 的获取并不容易，其需要利用一定的控制手段来实现 Φ_{imax}，其随燃料形式、燃烧结构、燃烧工艺等因素的变化而不同，对于确定的燃料、锅炉型式及燃烧工艺，采用适当的控制策略，Φ_{imax}方可保证。

正是由于燃煤锅炉存在上述两个基本问题，长期以来，对于中小型燃煤锅炉的最佳控制问题，采用数字仿真试验的方法一直未能有效地解决控制器的最佳设计问题，更多的情况是在实际工作中，采用人工智能控制策略实现燃煤锅炉的自动控制。

五、三相电压型 PWM 整流器系统控制问题

（一）问题提出

随着电力电子变换器广泛应用于电能产生、变换、控制、输送和存储等领域，其在工业、航天航空、信息、交通、医疗、家电等行业具有重要作用。

众所周知，电力电子变换器（例如二极管不控整流电路和晶闸管相控整流电路）会对电网注入大量谐波与无功功率，造成严重的电网污染[64]。治理电网污染的根本措施，是要求变流装置借助于适当的控制，实现网侧电流正弦化且运行于单位功率因数，其控制系统结构如图 2-21 所示。

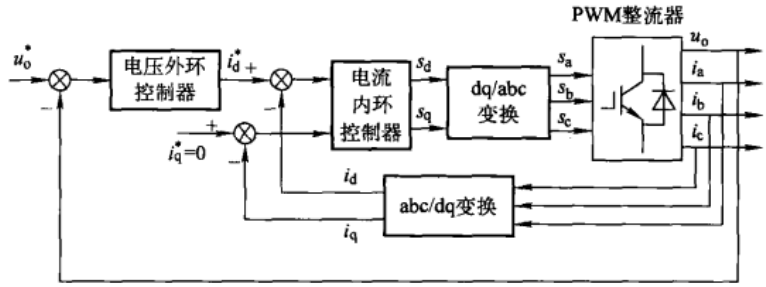


图 2-21　三相电压型 PWM 整流器控制系统结构图

图 2-21 所示的三相电压型 PWM 整流器可实现高功率因数、网侧电流正弦化、直流侧电压可调和能量可双向流动等诸多优点，目前已逐步取代不控或相控整流电路，在实际工程中得到了广泛应用。下面对三相电压型 PWM 整流器的建模问题展开论述，以便于对其控制系统的设计。

（二）建模机理

电力电子系统的建模问题可分为“系统建模”和“器件建模”两个方面的内容。

电力电子系统的建模机理主要有经典电路的基尔霍夫电压/电流定律和分析力学的“拉格朗日方程/哈密顿方程”[65]。其中，基尔霍夫电压/电流定律从电路细节处着眼，严格依赖于电力电子变换器的拓扑结构，不适用于复杂系统的建模。事实上，电力电子变换器系统在“微观上”表现为元件及其在拓扑结构上的约束关系，而在“宏观上”则表现为电场能、磁场能等系统能量。

因此，下面将从系统全局能量着眼，利用分析力学的基本原理——拉格朗日方程，建立三相电压型 PWM 整流器系统的数学模型，这种建模方法具有物理概念清晰、系统性强、适用于复杂电力电子变换器，以及便于从能量观点设计非线性控制器等优点。

分析力学是理论力学的一个分支，它通过选择广义坐标作为描述质点系运动的变量，运

用数学分析的方法，从系统能量的观点研究宏观现象中的力学问题。分析力学最初应用于机械系统的建模中（如前面介绍的龙门起重机建模问题），我们可通过类比分析，利用机械系统与电力电子系统物理量间的等效关系（表 2-2），将分析力学基本原理拓展应用于电力电子系统的建模中。

表 2-2 机械系统和电力电子系统物理量间的等效关系

机械系统	电力电子系统
x（质点位移）	q（电荷）
$\dot{x}$（质点速度）	$\dot{q}$（电流）
$\frac{1}{2}m\dot{x}^2$（质点动能）	$\frac{1}{2}L\dot{q}^2$（电感磁场能）
$-mgx$（重力势能）	$-qU$（电源电动势能）
$\frac{1}{2}kx^2$（弹性势能）	$\frac{1}{2C}q^2$（电容电动势能）
f（非有势力）	$R\dot{q}$（电阻电压）

对于电力电子系统，拉格朗日方程可表示为如下形式

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial T}{\partial \dot{q}_k}\right)-\frac{\partial T}{\partial q_k}=F_k \quad (k=1,2,\cdots,n) \tag{2-42}$$

式中，T 为系统磁场能量；q_k 为系统的广义坐标（电荷）；$\dot{q}_k$ 为系统的广义速度；n 为系统的自由度；F_k 为系统广义力，这里 $F_k=F_k^1+F_k^2$，F_k^1 为有势力，F_k^2 为非有势力，其具体表达式为

$$\begin{aligned} F_k^1&=-\frac{\partial V}{\partial q_k}\\ F_k^2&=-\frac{\partial D}{\partial \dot{q}_k}\end{aligned} \tag{2-43}$$

式中，V 为系统电场能量；D 为系统耗散能量[66,67]。

（三）系统建模

三相电压型 PWM 整流器的电路拓扑结构如图 2-22 所示。图中，e_a、e_b、e_c 为三相交流电源（不失一般性，这里假设三相电压不平衡），L 和 C 分别为滤波电感和滤波电容，R_1 是滤波电感的等效电阻，R_s 是开关管的等效电阻，则图中所示开关管可看做理想开关管。设总等效电阻 $R=R_1+R_s$，R_L 为负载电阻。设网侧三相交流电流分别为 i_a、i_b、i_c，整流电流为 i_{dc}，流过负载电阻的电流为 i_L，负载两端

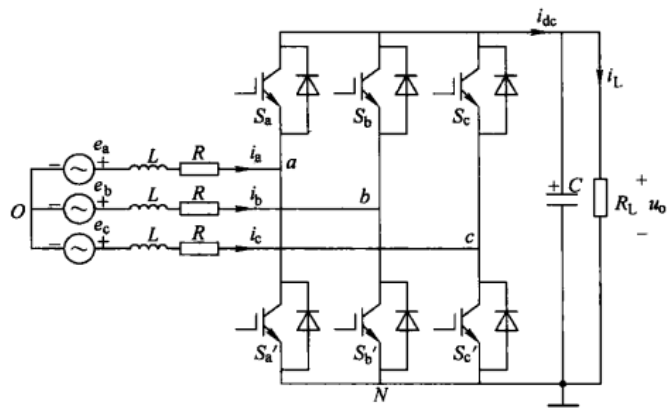


图 2-22 三相电压型 PWM 整流器

电压为 u_o。设三相交流电源的公共节点为 O，地线节点为 N。

定义二值逻辑开关函数 s_k 为

$$s_k = \begin{cases} 1 & \text{上桥臂导通，下桥臂关断} \\ 0 & \text{上桥臂关断，下桥臂导通} \end{cases} \quad (k = a,b,c)$$

（1）选取广义坐标　设交流侧三个电感的电荷量分别为 q_{La}、q_{Lb}、q_{Lc}，直流侧电容电荷量为 q_o，则 $i_a = \dot{q}_{La}$、$i_b = \dot{q}_{Lb}$、$i_c = \dot{q}_{Lc}$。由于

$$\dot{q}_{La} + \dot{q}_{Lb} + \dot{q}_{Lc} = 0 \tag{2-44}$$

对其两侧取积分，则得

$$q_{La} + q_{Lb} + q_{Lc} = Q \tag{2-45}$$

式中，Q 为常数。由此可见，q_{La}、q_{Lb}、q_{Lc}三者存在约束关系，只有其中的两个变量可以选取为广义坐标，这里选择 q_{La}和 q_{Lb}。因此，相互独立的 q_{La}、q_{Lb}、q_o 为三相电压型 PWM 整流器的广义坐标。

（2）计算系统能量函数　系统的磁场能量为

$$T = \frac{1}{2}L(\dot{q}_{La}^2 + \dot{q}_{Lb}^2 + \dot{q}_{Lc}^2) \tag{2-46}$$

系统的电场能量为

$$V = \frac{1}{2C}q_o^2 - e_a q_{La} - e_b q_{Lb} - e_c q_{Lc} \tag{2-47}$$

系统的耗散能量函数为

$$D = \frac{1}{2}R(\dot{q}_{La}^2 + \dot{q}_{Lb}^2 + \dot{q}_{Lc}^2) + \frac{1}{2}R_L[s_a\dot{q}_{La} + s_b\dot{q}_{Lb} + s_c\dot{q}_{Lc} - \dot{q}_o]^2 \tag{2-48}$$

（3）代入并整理拉格朗日方程　将以上系统能量函数代入式（2-42）所示的拉格朗日方程中，可得

$$\begin{cases} 2L\ddot{q}_{La} + L\ddot{q}_{Lb} - (e_a - e_c) = -2R\dot{q}_{La} - R\dot{q}_{Lb} - \dfrac{q_o}{C}(s_a - s_c) \\ 2L\ddot{q}_{Lb} + L\ddot{q}_{La} - (e_b - e_c) = -2R\dot{q}_{Lb} - R\dot{q}_{La} - \dfrac{q_o}{C}(s_b - s_c) \\ \dfrac{1}{C}q_o = R_L[(s_a - s_c)\dot{q}_{La} + (s_b - s_c)\dot{q}_{Lb} - \dot{q}_o] \end{cases} \tag{2-49}$$

参照 a、b 两相方程形式，由对称性可给出 c 相方程，进一步整理可得三相电网不平衡条件下 PWM 整流器系统的数学模型为

$$\begin{cases} L\dfrac{di_a}{dt} = -Ri_a - \dfrac{u_o}{3}(2s_a - s_b - s_c) + \dfrac{2e_a - e_b - e_c}{3} \\ L\dfrac{di_b}{dt} = -Ri_b - \dfrac{u_o}{3}(2s_b - s_a - s_c) + \dfrac{2e_b - e_a - e_c}{3} \\ L\dfrac{di_c}{dt} = -Ri_c - \dfrac{u_o}{3}(2s_c - s_a - s_b) + \dfrac{2e_c - e_a - e_b}{3} \\ C\dfrac{du_o}{dt} = s_a i_a + s_b i_b + s_c i_c - \dfrac{u_o}{R_L} \end{cases} \tag{2-50}$$

若三相平衡，即

$$e_a + e_b + e_c = 0 \tag{2-51}$$

则可得三相电网电压平衡时的 PWM 整流器数学模型为

$$\begin{cases} L\dfrac{di_a}{dt} = -Ri_a - u_o\left(s_a - \dfrac{1}{3}\sum\limits_{k=a,b,c} s_k\right) + e_a \\ L\dfrac{di_b}{dt} = -Ri_b - u_o\left(s_b - \dfrac{1}{3}\sum\limits_{k=a,b,c} s_k\right) + e_b \\ L\dfrac{di_c}{dt} = -Ri_c - u_o\left(s_c - \dfrac{1}{3}\sum\limits_{k=a,b,c} s_k\right) + e_c \\ C\dfrac{du_o}{dt} = s_a i_a + s_b i_b + s_c i_c - \dfrac{u_o}{R_L} \end{cases} \tag{2-52}$$

由式（2-52）可知，基于拉格朗日方程所建立的 PWM 整流器数学模型与参考文献[64，68]中采用基尔霍夫电压/电流定律所得模型是完全一致的。然而，与基尔霍夫定律相比，拉格朗日方程建模方法具有物理概念清晰、系统性强、适用于复杂电力电子系统建模等优点。

建议：感兴趣的读者可以采用基尔霍夫电压/电流定律，自行推导出 PWM 整流器系统数学模型，以此来体会这两种建模方法的不同之处。

（四）模型变换

由式（2-50）和式（2-52）知，三相电压型 PWM 整流器的数学模型为非线性微分方程组，为便于应用经典控制理论对控制系统进行设计，需要将其变换为传递函数形式的系统数学模型。

同时，由于模型中三相电压、电流均为交流量，不利于对其进行小信号线性化处理，因此首先需要通过坐标变换将其变为直流量，即进行 abc/dq 坐标变换，以利于系统的分析与设计。

在电网平衡情况下，三相 abc 静止坐标系到两相 dq 同步旋转坐标系的“等功率”坐标变换矩阵为

$$\boldsymbol{C}_{3s/2r} = \sqrt{\frac{2}{3}}\begin{bmatrix} \cos\theta & \cos(\theta-120°) & \cos(\theta+120°) \\ -\sin\theta & -\sin(\theta-120°) & -\sin(\theta+120°) \\ \dfrac{\sqrt{2}}{2} & \dfrac{\sqrt{2}}{2} & \dfrac{\sqrt{2}}{2} \end{bmatrix} \tag{2-53}$$

式中，$\theta=\omega t$，ω 为工频角频率，t 为时间。$\boldsymbol{C}_{3s/2r}$的逆矩阵为

$$\boldsymbol{C}_{2r/3s} = \sqrt{\frac{2}{3}}\begin{bmatrix} \cos\theta & -\sin\theta & \dfrac{\sqrt{2}}{2} \\ \cos(\theta-120°) & -\sin(\theta-120°) & \dfrac{\sqrt{2}}{2} \\ \cos(\theta+120°) & -\sin(\theta+120°) & \dfrac{\sqrt{2}}{2} \end{bmatrix} \tag{2-54}$$

将上述坐标变换代入式（2-52）中，则可得 dq 坐标系下 PWM 整流器数学模型为

$$\begin{cases} L\dfrac{\mathrm{d}i_\mathrm{d}}{\mathrm{d}t} = -Ri_\mathrm{d} + \omega Li_\mathrm{q} - s_\mathrm{d}u_\mathrm{o} + e_\mathrm{d} \\ L\dfrac{\mathrm{d}i_\mathrm{q}}{\mathrm{d}t} = -\omega Li_\mathrm{d} - Ri_\mathrm{q} - s_\mathrm{q}u_\mathrm{o} + e_\mathrm{q} \\ C\dfrac{\mathrm{d}u_\mathrm{o}}{\mathrm{d}t} = s_\mathrm{d}i_\mathrm{d} + s_\mathrm{q}i_\mathrm{q} - \dfrac{u_\mathrm{o}}{R_\mathrm{L}} \end{cases} \tag{2-55}$$

式中，i_d、i_q 分别是三相电流在同步旋转坐标系下的 d 轴分量和 q 轴分量；e_d、e_q 分别是三相交流电压在同步旋转坐标系下的 d 轴分量和 q 轴分量；s_d、s_q 分别是开关函数在同步旋转坐标系下的 d 轴分量和 q 轴分量。

为对式（2-55）所示系统数学模型进行小信号处理，可将其方程中各变量等效为稳态值和小信号扰动值之和的形式，即令 $u_\mathrm{o} = U_\mathrm{o} + \hat{u}_\mathrm{o}$，$i_\mathrm{d} = I_\mathrm{d} + \hat{i}_\mathrm{d}$，$i_\mathrm{q} = I_\mathrm{q} + \hat{i}_\mathrm{q}$，$s_\mathrm{d} = S_\mathrm{d} + \hat{s}_\mathrm{d}$，$s_\mathrm{q} = S_\mathrm{q} + \hat{s}_\mathrm{q}$。其中，$U_\mathrm{o}$、$I_\mathrm{d}$、$I_\mathrm{q}$、$s_\mathrm{d}$、$s_\mathrm{q}$ 为稳态值；$\hat{u}_\mathrm{o}$、$\hat{i}_\mathrm{d}$、$\hat{i}_\mathrm{q}$、$\hat{s}_\mathrm{d}$、$\hat{s}_\mathrm{q}$ 为小信号扰动值。将这些变量代入式（2-55）中，忽略高阶微分变量，可得“微分方程”形式描述的系统小信号模型为

$$\begin{cases} L\dfrac{\mathrm{d}\hat{i}_\mathrm{d}}{\mathrm{d}t} = -R\hat{i}_\mathrm{d} + \omega L\hat{i}_\mathrm{q} - S_\mathrm{d}\hat{u}_\mathrm{o} - U_\mathrm{o}\hat{s}_\mathrm{d} \\ L\dfrac{\mathrm{d}\hat{i}_\mathrm{q}}{\mathrm{d}t} = -\omega L\hat{i}_\mathrm{d} - R\hat{i}_\mathrm{q} - S_\mathrm{q}\hat{u}_\mathrm{o} - U_\mathrm{o}\hat{s}_q \\ C\dfrac{\mathrm{d}\hat{u}_\mathrm{o}}{\mathrm{d}t} = S_\mathrm{d}\hat{i}_\mathrm{d} + S_\mathrm{q}\hat{i}_\mathrm{q} + I_\mathrm{d}\hat{s}_\mathrm{d} + I_\mathrm{q}\hat{s}_q - \dfrac{\hat{u}_\mathrm{o}}{R_\mathrm{L}} \end{cases} \tag{2-56}$$

由于稳态时无功电流 $i_\mathrm{q}=0$，并对式（2-56）小信号模型进行拉普拉斯变换，可得

$$\begin{cases} LsI_\mathrm{d}(s) = -RI_\mathrm{d}(s) - S_\mathrm{d}U_\mathrm{o}(s) - U_\mathrm{o}S_\mathrm{d}(s) \\ CsU_\mathrm{o}(s) = S_\mathrm{d}I_\mathrm{d}(s) + I_\mathrm{d}S_\mathrm{d}(s) - \dfrac{1}{R_\mathrm{L}}U_\mathrm{o}(s) \end{cases} \tag{2-57}$$

由式（2-57）进一步可得如下的 PWM 整流器“控制输入-直流输出”传递函数为

$$G(s) = \frac{U_\mathrm{o}(s)}{S_\mathrm{d}(S)} = K\frac{\tau s + 1}{s^2 + 2\zeta\omega_\mathrm{n}s + \omega_\mathrm{n}^2} \tag{2-58}$$

式中，$K = \dfrac{RI_\mathrm{d} - U_\mathrm{o}S_\mathrm{d}}{LC}$；$\tau = \dfrac{LI_\mathrm{d}}{RI_\mathrm{d} - U_\mathrm{o}S_\mathrm{d}}$；$\zeta = \dfrac{L + R_\mathrm{L}RC}{2\sqrt{R_\mathrm{L}LC\ (R + R_\mathrm{L}S_\mathrm{d}^2)}}$；$\omega_\mathrm{n} = \sqrt{\dfrac{R + R_\mathrm{L}S_\mathrm{d}^2}{R_\mathrm{L}LC}}$。

该传递函数形式可用图 2-23 所示的 PWM 整流器系统动态结构图来表示。

（五）模型分析

由式（2-58）可见，系统开环传递函数的零点为 $-\dfrac{1}{\tau}$，由于在正常工作时 $\tau<0$（$U_\mathrm{o}S_\mathrm{d} \gg RI_\mathrm{d}$），故开环传递函数的零点在复平面右半平面。因此，三相电压型 PWM 整流器系统在正常工作时对系统输出电压呈现“非最小相位特性”。

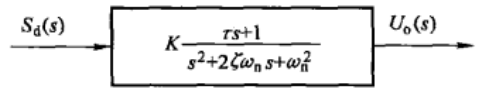


图 2-23　PWM 整流器系统动态结构图

仿真实验表明，三相电压型 PWM 整流器的“非最小相位特性”将导致系统输出电压阶跃响应具有“负调”现象（图 2-24），即输出电压在阶跃响应的初始阶段有“下调现象”。

这种“负调”现象可以直观解释为：在控制量变化的初始时刻，系统电感储能要增大

（减小），短时间内减少（增多）了送往电容的能量，从而导致电容电压在初始时刻减小（增大）。

这种非最小相位特性及其所致的“负调”现象，在其他 Boost 型开关变换器中也普遍存在。所以，明晰三相电压型 PWM 整流器的非最小相位特性对系统控制器设计具有重要的指导意义[70-72]。

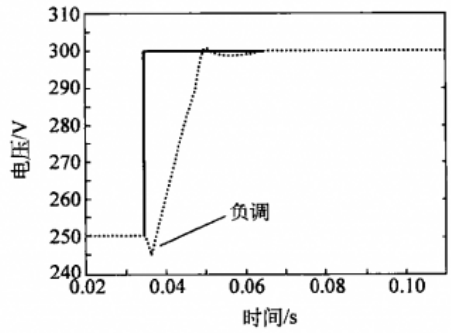


图 2-24 输出电压阶跃响应的“负调”现象

（六）模型验证

三相电压型 PWM 整流器系统的模型验证问题请读者参见本书第五章第五节中的相关内容。

六、磁悬浮轴承运动控制问题

（一）问题提出

磁悬浮轴承也称电磁轴承或磁力轴承，是新一代的非接触支承部件，已广泛地应用于空间技术、机械加工、机器人等众多领域，其实物外形如图 2-25 所示。它利用磁场力将轴杆无机械摩擦、无润滑地悬浮在空间中，是磁悬浮原理在动力机械领域中的一个典型应用案例。磁悬浮轴承具有传统轴承无法比拟的许多优越性能，如容许转子达到很高的转速，轴承功耗小，转子与定子之间可实现无摩擦的相对运动，维护成本低，寿命长等。

虽然相比传统机械轴承来说，磁悬浮轴承具有诸多优势，但它也存在一些工程实际问题。例如，它的固有特性在本质上存在较强的非线性，难以实现有效的控制。由于磁悬浮轴承是由“多组电磁线圈 + 动态平衡控制”构成的复杂的磁机电复合控制系统，其良好的动态性能是以“控制”为基础的。因此，需要首先建立该系统的数学模型，以便于今后对其控制系统的设计。

（二）建模机理[48]

为研究磁悬浮轴承的运动控制问题，需要对实际系统进行简化与抽象。下面从一组电磁线圈（垂直方向）入手，对磁悬浮轴承在垂直方向“电磁力/电流-轴杆位置”的动态特性进行分析，其简化的物理模型如图 2-26 所示。

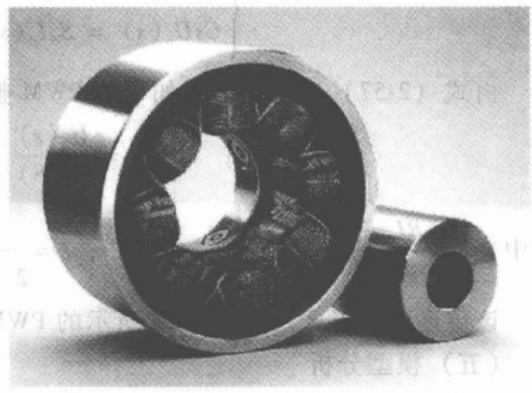


图 2-25 磁悬浮轴承

图 2-26 中将轴杆抽象成一个匀质金属小棒，起动电磁铁时，轴杆受到上方电磁力的吸引而克服重力的作用，实现磁悬浮。可以看出，此问题为单自由度问题，因而依据经典力学的牛顿定律即可求解。

（三）系统建模

我们知道，实际系统常常会受到不确定性因素的干扰，如空气阻力、磁场复杂性、转轴快速旋转时产生的陀螺效应等。为了便于建模分析，需要忽略一些对系统建模的次要影响，提取出影响系统性态的最主要因素。据此，可对实际系统进一步抽象得到如图 2-27 所示的物理模型。

电磁铁线圈通入电流 i(A)，其匝数为 N。金属轴杆质量 m(kg)，磁通面积 S(m^2)，磁感应

强度 B(T),电磁铁北、南两极产生的电磁力分别为 F_1(N)和 F_2(N),金属轴杆在电磁合力与重力 mg(N)的共同作用下在垂直方向运动,轴杆与电磁铁距离 x(m)。平衡距离 x_0(m),平衡电流 i_0(A)。空气磁导率 μ_0(N/A^2),电磁铁及轴杆的磁导率 μ_r(N/A^2)。其他的扰动忽略。

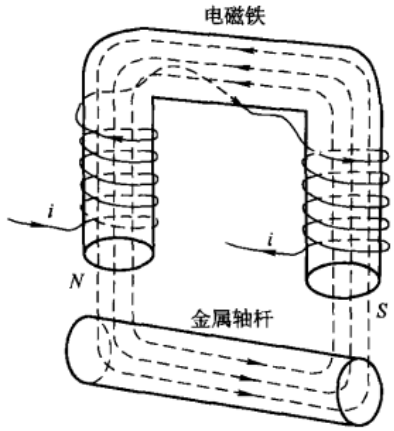


图 2-26　磁悬浮轴承的简化物理模型

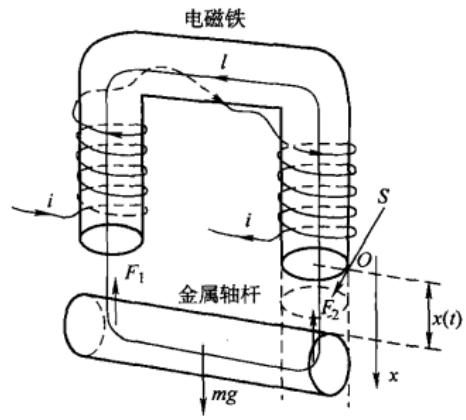


图 2-27　磁悬浮轴承物理模型

为了推导建立系统的数学模型,这里需要引用以下几个电磁学定理(具体的证明过程可以参照参考文献[73])。

(1) 磁场能量体密度公式　设空间中某点的磁感应强度为 B(T),磁场强度为 H(A/m),那么此点的磁场能量体密度为 $w_m = \frac{1}{2}HB$。

(2) 磁场力公式　对于一个磁场能量为 W_m (J) 的磁场区域，若外电路对其提供的能量保持不变，区域内仅有一个广义坐标 g 发生变化，则此区域产生的广义磁场力为

$$F = -\frac{\mathrm{d}W_m}{\mathrm{d}g}$$

(3) 安培环路定理　在图 2-27 所示的磁场中任意取一闭合路径 l，则磁感应强度沿此回路的线积分应为

$$\oint_l H\mathrm{d}l = \sum I_k$$

式中，$\sum I_k$ 为所有穿过该回路所包围面积的自由电流的代数和。

利用上述引理，建模过程如下：

根据磁场能量体密度公式知，电磁铁所在区域 V 内的总磁场能量为

$$W_m = \int_V w_m \mathrm{d}V = \int_V \frac{1}{2}HB\mathrm{d}V = \frac{1}{2}\int_V HB\mathrm{d}V = \frac{1}{2}\int_V \frac{B^2}{\mu}\mathrm{d}V \tag{2-59}$$

由于电磁铁的钢心内部磁导率 μ_r 很大，所以磁场强度很小，故储存在铁磁媒质中的磁场能量远小于储存在空气气隙中的部分，因此前者可以忽略不计。分析图 2-27 可知，储存在每个气隙中的磁场能量为

$$W'_m = \int_{V'} w_m \mathrm{d}V = \int_{V'} \frac{1}{2}HB\mathrm{d}V = \frac{1}{2}\int_{V'} HB\mathrm{d}V = \frac{1}{2}\int_{V'} \frac{B^2}{\mu}\mathrm{d}V = \frac{1}{2}\frac{B^2}{\mu_0}Sx \tag{2-60}$$

所以磁悬浮轴承系统的总磁场能量为

$$W_m = 2W'_m = \frac{B^2 Sx}{\mu_0} \tag{2-61}$$

设通入电磁铁线圈的电流 i 相对时间为恒值，则外电路对系统提供的能量可视为不变。以轴杆与电磁铁的距离 x 作为广义坐标，因为电磁引力有使 x 减小的趋势，所以根据磁场力公式得电磁铁对轴的电磁引力为

$$F_1 + F_2 = F = -\frac{\mathrm{d}W_m}{\mathrm{d}x} = -\frac{\mathrm{d}}{\mathrm{d}x}\left(\frac{B^2 Sx}{\mu_0}\right) = -\frac{B^2 S}{\mu_0} \tag{2-62}$$

取力的方向为垂直向上，则负号可以去掉。

取一条磁感线回路作为积分回路 l，由安培环路定理得

$$Ni = \sum i_k = \oint_l H\mathrm{d}l = \oint_l \frac{B}{\mu}\mathrm{d}l = \oint_{l-2x} \frac{B}{\mu_0\mu_r}\mathrm{d}l + \oint_{2x} \frac{B}{\mu_0}\mathrm{d}l \tag{2-63}$$

因为 μ_r 很大，因此第一项可忽略，简化后得

$$Ni = \oint_{2x} \frac{B}{\mu_0}\mathrm{d}l = \frac{2Bx}{\mu_0} \tag{2-64}$$

则

$$B = \frac{Ni\mu_0}{2x} \tag{2-65}$$

代入式（2-62）得

$$F = \frac{B^2 S}{\mu_0} = \left(\frac{Ni\mu_0}{2x}\right)^2 \frac{S}{\mu_0} = \frac{\mu_0 SN^2 i^2}{4x^2} \tag{2-66}$$

由图 2-27 可知，轴杆受到的合外力为 $mg - F$，再由牛顿第二定律，得到系统的表达式为

$$m\ddot{x} = mg - F = mg - \frac{\mu_0 SN^2 i^2}{4x^2} \tag{2-67}$$

从式（2-67）可以看出，磁悬浮轴承系统特性是非线性的。当距离 x 很小时，等式右端的第二项很大，即轴杆主要受到电磁引力的作用，表现为轴杆被吸引而向上运动。当距离 x 很大时，等式右端的第二项很小，即轴杆主要受到重力的作用，表现为轴杆向下坠落。这与我们的直观印象相符，间接证明了数学模型的正确性。

（四）模型简化

由式（2-65）可见，磁悬浮轴承的运动控制方程是一个非线性微分方程。为便于应用经典控制理论进行控制系统设计，需将其在平衡工作点附近进行线性化。

电磁引力 F 由线圈电流 $i(t)$ 和气隙距离 $x(t)$ 这两个变量决定，假定轴杆移动的距离远小于平衡距离 x_0，应用参考文献［49］中的小信号线性化方法，则式（2-66）可简化为

$$F(x,i) = F(x = x_0, i = i_0) + k_1(i(t) - i_0) + k_2(x(t) - x_0) \tag{2-68}$$

式中，系数 k_1 和 k_2 为

$$k_1 = \left[\frac{\partial F}{\partial i}\right]_{x_0,i_0} = \left[\frac{\partial}{\partial i}\left(\frac{\mu_0 SN^2 i^2}{4x^2}\right)\right]_{x_0,i_0} = \frac{\mu_0 SN^2 i_0}{2x_0^2} \tag{2-69}$$

$$k_2 = \left[\frac{\partial F}{\partial x}\right]_{x_0,i_0} = \left[\frac{\partial}{\partial x}\left(\frac{\mu_0 SN^2 i^2}{4x^2}\right)\right]_{x_0,i_0} = -\frac{\mu_0 SN^2 i_0^2}{2x_0^3} \tag{2-70}$$

已知在平衡工作点处有

$$F(x = x_0, i = i_0) = mg \tag{2-71}$$

代入式(2-67)，得系统的线性化模型为

$$m\ddot{x} = -\frac{\mu_0 SN^2 i_0}{2x_0^2} i + \frac{\mu_0 SN^2 i_0^2}{2x_0^3} x \tag{2-72}$$

对磁悬浮轴承的线性化模型式(2-72)进行拉普拉斯变换，可以得到距离与电流之间的传递函数表达式，即

$$G(s) = \frac{X(s)}{I(s)} = \frac{-\dfrac{\mu_0 SN^2 i_0}{2mx_0^2}}{s^2 - \dfrac{\mu_0 SN^2 i_0^2}{2mx_0^3}} = \frac{-2ab}{(s+a)(s-a)} \tag{2-73}$$

其中

$$a = \frac{Ni_0}{x_0}\sqrt{\frac{\mu_0 S}{2mx_0}},\quad b = \frac{N}{2}\sqrt{\frac{\mu_0 S}{2mx_0}}$$

据此可得到系统的动态结构，如图 2-28 所示。

从传递函数可以看出，磁悬浮轴承系统在 s 平面的右半部分有一个极点 $\lambda = a$，即磁悬浮轴承系统是一个“非最小相位系统”，所以必须施加一定的控制才能使系统保持稳定状态。

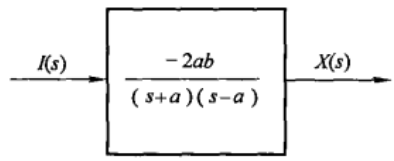


图 2-28　磁悬浮轴承系统动态结构图

同理可求得系统状态空间方程如下

$$\begin{cases} \dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}u \\ \boldsymbol{y} = \boldsymbol{C}\boldsymbol{x} \end{cases} \tag{2-74}$$

式中，$\boldsymbol{x} = [x, \dot{x}]^{\mathrm{T}}$；$u = i$；$\boldsymbol{y} = x$；

$$\boldsymbol{A} = \begin{bmatrix} 0 & 1 \\ \dfrac{\mu_0 SN^2 i_0^2}{2mx_0^3} & 0 \end{bmatrix},\quad \boldsymbol{B} = \begin{bmatrix} 0 \\ -\dfrac{\mu_0 SN^2 i_0}{2mx_0^2} \end{bmatrix},\quad \boldsymbol{C} = [1 \quad 0]。$$

以上就是仅考虑一组电磁线圈作用下，磁悬浮轴承的建模过程。如果要考虑多组电磁线圈结构情况下的整体建模问题，其过程将更为复杂。感兴趣的读者可以参照参考文献[74－76]，作进一步的分析。

（五）模型验证

（1）模型封装　在 Simulink 环境下建立系统仿真用的模型，如图 2-29 所示。模型中的上半部分为系统的精确模型，下半部分为系统的简化模型。

其中：

Fcn：u0 * s * (n^2) * (u[2]^2)/(m * (u[1]^2))

Fcn1：2 * u0 * s * (n^2) * u[1]/(m * (3.14^3))

（2）模型验证　采用必要条件法来验证所建立的数学模型是对实际系统的真实再现。

通过分析可知，磁悬浮轴承系统是含有一个不稳定平衡点的非最小相位系统。若初始状态时，位于 x_0 处的金属轴杆是静止的，那么通过微分方程可以解出保持轴杆平衡的工作电流 i_0。若对线圈施加的电流为大于 i_0 的恒值，轴杆会因受到大于重力的引力，而被吸到电磁

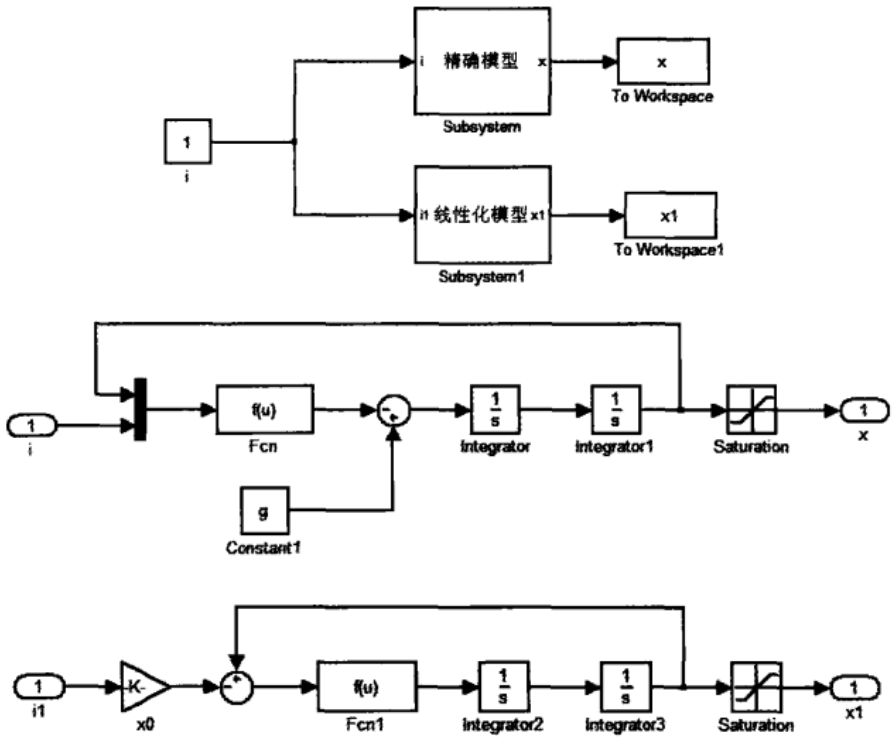


图 2-29 磁悬浮轴承系统模型框图

铁上。若对线圈施加的电流为小于 i_0 的恒值，轴杆会因受到的引力无法抵消重力的作用而远离电磁铁。基于这一点，可以分别使通入的电流大于、等于、小于工作电流，来观测系统的响应。下面利用仿真实验来验证这一模型的正确性。

为便于计算，这里的参数取：$m = 0.25\text{kg}$，$g = 9.8\text{m/s}^2$，$x_0 = 3.14\text{mm}$，$N = 10^4$，$S = 793.3\text{mm}^2$，$\pi = 3.14$，仿真结果如图 2-30 所示。

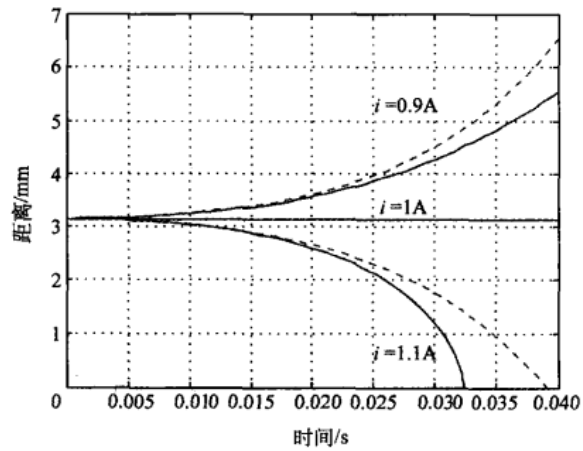


图 2-30 电流取不同恒值时系统响应曲线

图中，实线是精确模型的响应曲线，虚线是线性化模型的响应曲线。可以看出，当通入线圈的电流等于工作电流，即 1A 时，金属轴杆与电磁铁的距离保持恒定，此时即为系统的不稳定平衡状态。当通入线圈的电流等于 1.1A 时，轴杆受到电磁铁的吸引而向电磁铁运动，直到吸附在电磁铁上。当通入线圈的电流等于 0.9A 时，轴杆在重力作用下，远离电磁铁，距离以近似于时间的二次函数的方式增长。另外，在工作距离附近的较小的区域内，线性化模型与精确模型的重合度较好。

因此，可以认为线性化模型在一定条件下可以表述原系统模型的性质。

第三节　实现问题

控制系统数学模型的建立（称之为一次模型化），为进行系统仿真实验研究提供了必要的前提条件，但真正在数字计算机上对系统模型实现仿真运算、分析，还有一个关键步骤，就是所谓“实现问题”。

在控制理论中，所谓“实现问题”就是根据已知的系统传递函数求取该系统相应的状态空间表达式，也就是说，把系统的外部模型（传递函数描述）形式转化为系统的内部模型（状态空间描述）形式。这对于计算机仿真技术而言，是一个具有实际意义的问题。因为状态方程是一阶微分方程组形式，非常适宜用数字计算机求其数值解（而高阶微分方程的数值求解是非常困难的）。如果控制系统已表示为状态空间表达式，则很容易直接对该表达式编制相应的求解程序。

对一个已知的系统传递函数，其相应的状态方程实现并不唯一，本节仅以单变量系统可控标准型实现作为一种方法加以说明，更深入的内容不在此讨论。

一、单变量系统的可控标准型实现

设系统传递函数为

$$G(s) = \frac{Y(s)}{U(s)} = \frac{c_1 s^{n-1} + \cdots + c_{n-1}s + c_n}{s^n + a_1 s^{n-1} + \cdots + a_{n-1}s + a_n}$$

若对上式设

$$\frac{Z(s)}{U(s)} = \frac{1}{s^n + a_1 s^{n-1} + \cdots + a_{n-1}s + a_n}$$

$$\frac{Y(s)}{Z(s)} = c_1 s^{n-1} + \cdots + c_{n-1}s + c_n$$

再经拉普拉斯反变换，有

$$z^{(n)}(t) + a_1 z^{(n-1)}(t) + \cdots + a_{n-1}z'(t) + a_n z(t) = u(t) \tag{2-75}$$

$$y(t) = c_1 z^{(n-1)}(t) + \cdots + c_{n-1}z'(t) + c_n z(t)$$

引入 n 维状态变量 $\boldsymbol{X} = [x_1, x_2, \cdots, x_n]$，并设

$$x_1 = z$$

$$x_2 = z' = x_1'$$

$$\vdots$$

$$x_n = z^{(n-1)} = x_{n-1}'$$

再由式（2-75）有

$$x_n' = z^{(n)} = -a_1 z^{(n-1)} - \cdots - a_{n-1}z' - a_n z + u = -a_1 x_n - \cdots - a_{n-1}x_2 - a_n x_1 + u \tag{2-76}$$

$$y = c_n x_1 + c_{n-1}x_2 + \cdots + c_1 x_n \tag{2-77}$$

得到一阶微分方程组

$$\begin{cases} x_1' = x_2 \\ x_2' = x_3 \\ \quad \vdots \\ x_{n-1}' = x_n \\ x_n' = -a_n x_1 - \cdots - a_2 x_{n-1} - a_1 x_n + u \end{cases}$$

写为矩阵形式为

$$\begin{cases} \dot{\boldsymbol{X}} = \boldsymbol{AX} + \boldsymbol{BU} \\ \boldsymbol{Y} = \boldsymbol{CX} + \boldsymbol{DU} \end{cases}$$

就得到了系统的内部模型描述——状态空间表达式。其中

$$\boldsymbol{A} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & \cdots & 1 \\ -a_n & -a_{n-1} & \cdots & \cdots & -a_1 \end{bmatrix}; \quad \boldsymbol{B} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}$$

$$\boldsymbol{C} = \begin{bmatrix} c_n & c_{n-1} & \cdots & \cdots & c_1 \end{bmatrix}; \quad \boldsymbol{D} = [0]$$

其一阶微分矩阵向量形式很便于在计算机上运用各种数值积分方法求取数值解（下节将予以详细阐述）。

将系统的状态方程描述式（2-76）和式（2-77）用图形方式表示如图 2-31 所示。

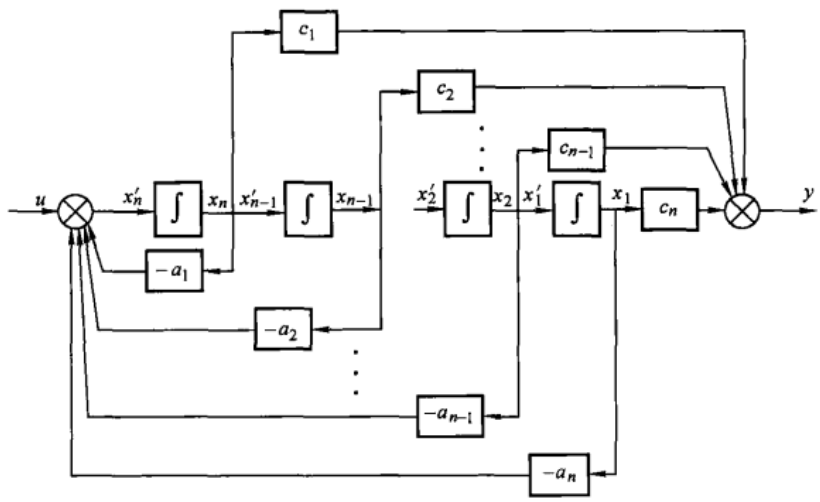


图 2-31 单变量系统的可控标准型模拟实现图

图 2-31 清楚地表明了系统内部状态变量之间的相互关系和内部结构形式，通常称为模拟实现图。从图中可知，欲知各状态变量 x_1，x_2，…，x_n 的动态特性变化情况，对于数字计算机来讲关键在于求解各状态变量的一阶微分 x_1'，x_2'，…，x_n'。因此，图中各积分环节的作用至关重要。采用传统的模拟计算机求解，则积分环节由运算放大器构成的积分器实现；而采用数字计算机求解，积分环节由各种数值积分算法实现。可以说模拟实现图给出了清晰的系统仿真模型。

二、控制系统的数字仿真实现

控制系统计算机仿真技术所要求的“实现问题”更为具体，是指如何将已得到的控制系统的数学模型通过一定方法、手段转换为可在数字计算机上运行求解的“仿真模型”，实际上是“仿真实现”问题，或称做“二次模型化”过程。“控制系统数字仿真与 CAD”这门课程很重要的一部分内容就是研究二次模型化问题，即如何建立控制系统仿真模型，使其在数字计算机上得到“实现”，进而求解运算，得到所需的运行结果。这也是仿真领域长期以来一直进行的重点研究工作。

一般地说，控制系统的数字仿真实现有以下几个步骤：

1）根据已建立的数学模型和精度、计算时间等要求，确定所采用的数值计算方法。

2）将原模型按照算法要求通过分解、综合、等效变换等方法转换为适于在数字计算机上运行的公式、方程等。

3）用适当的软件语言将其描述为数字计算机可接受的软件程序，即编程实现。

4）通过在数字计算机上运行，加以校核，使之正确反映系统各变量动态性能，得到可靠的仿真结果。

围绕以上步骤，系统仿真技术近年来不断发展、不断更新，各类控制系统专用仿真软件为适应仿真中的二次模型化需求不断推出。其中最具特色的就是美国学者 Cleve Moler 等人于 1980 年推出的交互式 MATLAB 语言。在此基础上，陆续出现的许多专门用于控制系统分析与 CAD 的工具箱，对系统仿真技术的发展起到很大的推动作用。

一个良好的算法软件，如 MATLAB 语言，可以使系统仿真研究人员把精力集中于仿真模型的建立和求解方法的确定、仿真结果的分析和控制系统的设计这类重要和关键问题上来。而对于采用什么算法，如何保证精度，如何逐条编程实现这样一些底层问题不必花费过多的心思，不必去详细了解相应算法的一些具体内容，从而提高工作效率，并保证了软件的可靠性。因此，高水平的算法软件的出现，使得原本复杂艰巨的二次模型化任务变得容易了，也就是说，仿真的实现问题在强大功能软件的支持下，能够很方便地得到解决。这一点在以后章节中将给予详细阐述。

第四节　常微分方程数值解法

控制系统数学模型经合理近似、简化，大多建立成为常微分方程形式。实际中遇到的大部分微分方程难以得到解析解，通常都是通过数字计算机采用数值计算方法求取数值解。尽管在高级仿真软件（MATLAB）环境下，已提供了功能十分强大，且能保证相应精度的数值求解的功能函数或程序段，使用者仅需按规定的语言规格调用即可，而无需从数值算法的底层考虑其编程实现过程，但为掌握数字仿真技术的基本技能，在仿真分析和设计中合理选择和使用相应的算法以获得满足要求的数值结果，有必要对常微分方程的数值求解问题作较深入的了解。本节将从数值求解的基本概念入手，介绍系统仿真中常用的几种数值求解方法及其使用特点。

一、数值求解的基本概念

控制系统的数学模型可能是状态方程描述，也可能是传递函数描述，或其他微分方程组形式描述，但均可以通过“实现”的方法化为一阶微分方程组的形式来求解，故本节主要以一阶微分方程为基础来讨论数值求解的基本概念。

设常微分方程为

$$\begin{cases}\dfrac{dy}{dt}=f(t,y)\\ y(t_0)=y_0\end{cases} \tag{2-78}$$

则求解方程中函数 $y(t)$ 问题称为常微分方程初值问题。所谓数值求解就是要在时间区间 $[a、b]$ 中取若干离散点 $t_k(k=0,1,2,\cdots,N)$，且

$$a=t_0<t_1<\cdots<t_N=b$$

设法求出式（2-78）的解函数 $y(t)$ 在这些时刻上的近似值 $y_0,y_1,\cdots,y_N$，即求取

$$y_k\approx y(t_k),k=0,1,2,\cdots,N$$

从上可知，常微分方程数值解法的基本出发点就是离散化。即将连续时间求解区间 $[a,b]$ 分成若干离散时刻点 t_k，然后直接求出各离散点上的解函数 $y(t_k)$ 的近似值 y_k，而不必求出解函数 $y(t)$ 的解析表达式。这在一般工程实际中已满足大部分控制系统的仿真分析要求。

通常取求解区间 $[a,b]$ 的等分点作为离散点较方便，即设

$$y_k\approx y(t_k),k=0,1,2,\cdots,N$$

而令 $h=(b-a)/N$，称为等间隔时间步长。

求常微分方程数值解的基本方法有以下几种：

（一）差商法

设式（2-78）中的导数 y' 在 $t=t_k$ 处可用差分形式近似替代，即

$$y'(t_k)\approx\frac{y_{k+1}-y_k}{h} \tag{2-79}$$

则式（2-78）转化为

$$\begin{cases}\dfrac{y_{k+1}-y_k}{h}=f(t_k,y_k),k=0,1,\cdots,N-1\\ y_0=y(t_0)\end{cases} \tag{2-80}$$

显然由此可得微分方程初值问题的数值解序列值

$$\begin{cases}y_0=y(t_0)\\ y_{k+1}=y_k+hf(t_k,y_k),k=0,1,\cdots,N-1\end{cases} \tag{2-81}$$

呈现出递推关系，只要已知初值 $y(t_0)$，即可求得所需数值序列 $y_k(k=1,2,\cdots,N)$。

（二）台劳（Taylor）展开法

解函数 $y(t)$ 在 t_k 附近可展为台劳多项式

$$y(t_k+h)\approx y(t_k)+hy'(t_k)+\frac{h^2}{2!}y''(t_k)+\cdots+\frac{h^n}{n!}y^{(n)}(t_k)+\cdots$$

由式(2-78)、式(2-79)知

$$y(t_k+h)=y(t_{k+1})=y_{k+1}$$

并记

$$y'(t_k)=f(t_k,y_k)=f_k=y_k'$$

$$y''(t_k)=f_t'(t_k,y_k)+f_y(t_k,y_k)y'(t_k)=f_{t_k}'+f_{y_k}'f_k=y_k''$$

则式(2-81)可化为

$$\begin{cases}y_0=y(t_0)\\ y_{k+1}=y_k+hy_k'+\dfrac{h^2}{2!}y_k''+\cdots+\dfrac{h^n}{n!}y_k^{(n)}+\cdots\end{cases}\tag{2-82}$$

按求解精度要求，取适当项数 n，即可递推求解。特别地，当 $n=1$，则有

$$\begin{cases}y_0=y(t_0)\\ y_{k+1}=y_k+hy_k'=y_k+hf(t_k,y_k),k=0,1,\cdots,N-1\end{cases}$$

与式（2-81）完全相同。

（三）数值积分法

将式（2-78）在小区间［t_k，t_{k+1}］上积分，得

$$y_{k+1}=y_k+\int_{t_k}^{t_{k+1}}f(t,y)\,\mathrm{d}t\tag{2-83}$$

于是在区间［a，b］上，式（2-78）可化为

$$\begin{cases}y_0=y(t_0)\\ y_{k+1}=y_k+\displaystyle\int_{t_k}^{t_{k+1}}f(t,y)\,\mathrm{d}t,k=0,1,2,\cdots,N-1\end{cases}$$

只要对其中积分项采用数值积分方法求得即可。而数值积分的方式方法非常之多，需要根据仿真精度和计算时间要求来确定使用何种方法。

二、数值积分法

下面讨论控制系统仿真中最常用和最基本的数值积分法，并根据其特点，提供求解常微分方程初值问题时正确选用数值算法的参考依据。

（一）欧拉（Euler）法

设一阶微分方程如式（2-78），重写为

$$\frac{\mathrm{d}y}{\mathrm{d}t}=f(t,y)$$

初始条件：

$$y(t_0)=y_0$$

在［t_k，t_{k+1}］区间上积分，由式（2-83），得

$$y_{k+1}-y_k=\int_{t_k}^{t_{k+1}}f(t,y)\,\mathrm{d}t$$

又由导数定义知

$$\frac{\mathrm{d}y}{\mathrm{d}t}=\lim_{\Delta t\to 0}\frac{y(t+\Delta t)-y(t)}{\Delta t}$$

在 $t=t_k$ 时刻，取 $h=\Delta t=t_{k+1}-t_k$，则显然 $y_{k+1}=y(t+\Delta t)$，$y_k=y(t)$。设 h 足够小，使得

$$\frac{\mathrm{d}y}{\mathrm{d}t}=f(t_k,y_k)\approx\frac{y_{k+1}-y_k}{h} \tag{2-84}$$

成立，于是，由式（2-84）得

$$y_{k+1}-y_k=hf(t_k,y_k) \tag{2-85}$$

与式（2-83）比较，$hf(t_k,y_k)$ 部分近似代替了积分部分，即

$$\int_{t_k}^{t_{k+1}}f(t,y)\,\mathrm{d}t\approx hf(t_k,y_k) \tag{2-86}$$

其几何意义是把 $f(t,y)$ 在 $[t_k,\ t_{k+1}]$ 区间内的曲边面积用矩形面积近似代替，如图 2-32 所示。当 h 很小时，可以认为造成的误差是允许的。所以，式（2-83）就可写为

$$y_{k+1}=y_k+hf(t_k,y_k) \tag{2-87}$$

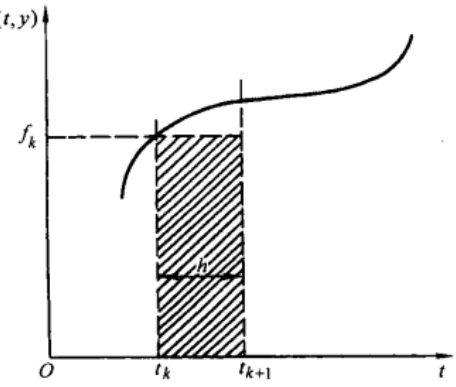


图 2-32 欧拉法的几何意义

取 $k=0,\ 1,\ 2,\ \cdots,\ N$，即可从 t_0 开始，逐点递推求得 t_1 时的 y_1，t_2 时的 y_2，$\cdots$，直至 t_N 时的 y_N，称之为欧拉递推公式，这也就是最简单的数值积分求解递推算法。

欧拉法方法简单，计算量小，由前一点值 y_k 仅一步递推就可以求出后一点值 y_{k+1}，属于单步法。又由于从初始值 y_0 即可开始进行递推运算，不需要其他信息，因此又属于自启动算法。

此处不难看出，欧拉法与泰勒展开式

$$y(t+\Delta t)=y(t)+y'(t)\Delta t+\frac{1}{2!}y''(t)(\Delta t)^2+\cdots \tag{2-88}$$

当 $t=t_k$，且取 $h=\Delta t$ 时，对应式

$$y_{k+1}=y_k+hy_k'+\frac{1}{2!}h^2y_k''+\cdots \tag{2-89}$$

中的一阶近似展开式相同，即

$$y_{k+1}=y_k+hy_k'+o(h^2)\approx y_k+hy_k'$$

其误差 $o(h^2)$ 与 h^2 同数量级，称其具有一阶精度，显然精度较差。尽管如此，欧拉法仍是非常重要的，许多高精度的数值积分方法都是以它为基础推导而得出的。

（二）龙格-库塔（Runge-Kutta）法

为使数值积分精度进一步提高，泰勒展开式取二阶近似式，则具有二阶精度，即截断误差 $o(h^3)$ 与 h^3 同数量级；若取四阶近似，则具有四阶精度，截断误差 $o(h^5)$ 更小，与 h^5 同数量级。但这使得求解 $f(t,y)$ 的高阶导数较为困难，这时通常可采用龙格-库塔法。龙格-库塔法的基本思路是：用函数值 $f(t,y)$ 的线性组合来代替 $f(t,y)$ 的高阶导数项，既可以避免计算高阶导数，又可以提高数值计算精度。其方法为

设 $y(t)$ 为式（2-78）的解，将其在 t_k 附近以 h 为变量展开为泰勒级数

$$y(t_k+h)=y(t_k)+hy'(t_k)+\frac{h^2}{2!}y''(t_k)+\cdots \tag{2-90}$$

因为

$$y'(t_k)=f(t_k,y_k)=f_k$$

$$y^{\cdot}(t_k)=\left.\frac{\mathrm{d}f(t,y)}{\mathrm{d}t}\right|_{\substack{t=t_k\\y=y_k}}=\left.\left(\frac{\partial f}{\partial t}+\frac{\partial f}{\partial y}f\right)\right|_{\substack{t=t_k\\y=y_k}}=f'_{t_k}+f'_{y_k}f_k$$

并记　$$y(t_k+h)=y_{k+1},\quad y(t_k)=y_k$$

于是　$$y_{k+1}=y_k+hf_k+\frac{h^2}{2!}(f'_{t_k}+f'_{y_k}f_k)+\cdots \tag{2-91}$$

上式中f'_{t_k}、f'_{y_k}等各阶导数不易计算，用下式中 k_i 的线性组合表示，则 y_{k+1}成为

$$y_{k+1}=y_k+h\sum_{i=1}^{r}b_ik_i \tag{2-92}$$

式中，r 为精度阶次；b_i 为待定系数，由所要求的精度确定；k_i 用下式表示

$$k_i=f\left(t_k+c_ih,y_k+h\sum_{j=1}^{i=1}a_jk_j\right),\qquad i=1,2,3,\cdots,r$$

式中，c_i、a_j 亦为待定系数，$j=1$，2，$\cdots$，$i-1$。一般均取 $c_1=0$。

当 $r=1$ 时，$k_1=f(t_k,y_k)$，则

$$y_{k+1}=y_k+hb_1k_1$$

与式（2-90）取一阶近似公式相比较，可得 $b_1=1$，则上式成为

$$y_{k+1}=y_k+hk_1$$
$$k_1=f(t_k,y_k)$$

此即欧拉法递推公式。其中 k_1 是 y_k 点的切线斜率。

当 $r=2$ 时，

$$k_1=f(t_k,y_k)$$
$$k_2=f(t_k+c_2h,y_k+ha_1k_1) \tag{2-93}$$

而 k_2 可按二元函数展开成为

$$k_2\approx f_k+c_2hf'_{t_k}+ha_1k_1f'_{y_k} \tag{2-94}$$

则
$$\begin{aligned}y_{k+1}&=y_k+b_1hk_1+b_2hk_2\\&=y_k+b_1hf_k+b_2h(f_k+c_2hf'_{t_k}+ha_1k_1f'_{y_k})\\&=y_k+(b_1+b_2)hf_k+b_2c_2h^2f'_{t_k}+a_1b_2h^2f_kf'_{y_k}\end{aligned}$$

同样与式（2-91）取二阶近似公式比较，即得以下关系式

$$b_1+b_2=1,\quad b_2c_2=\frac{1}{2},\quad a_1b_2=\frac{1}{2}$$

待定系数个数超过方程个数，必须先设定一个系数，然后即可求得其他系数。一般有以下几种取法：

1）$a_1=\frac{1}{2}$，$b_1=0$，$b_2=1$，$c_2=\frac{1}{2}$时，则

$$y_{k+1}=y_k+hk_2$$
$$k_1=f(t_k,y_k)$$
$$k_2=f\left(t_k+\frac{h}{2},y_k+\frac{h}{2}k_1\right)$$

2）$a_1=\frac{2}{3}$，$b_1=\frac{1}{4}$，$b_2=\frac{3}{4}$，$c_2=\frac{2}{3}$时，则

$$y_{k+1}=y_k+\frac{h}{4}(k_1+3k_2)$$

$$k_1=f(t_k,\ y_k)$$

$$k_2=f\left(t_k+\frac{2}{3}h,\ y_k+\frac{2}{3}hk_1\right)$$

3）$a_1=1$，$b_1=\frac{1}{2}$，$b_2=\frac{1}{2}$，$c_2=1$ 时，则

$$\begin{cases}y_{k+1}=y_k+\dfrac{h}{2}(k_1+k_2)\\ k_1=f(t_k,\ y_k)\\ k_2=f(t_k+h,\ y_k+hk_1)\end{cases}\tag{2-95}$$

以上几种递推公式均称为二阶龙格-库塔公式，是较典型的几个常用算法。其中 3）法又称为预估-校正法，或梯形法，意义如下：

用欧拉法以斜率 k_1 先求取一点 $\bar{y}_{k+1}$，称为预估点

$$\bar{y}_{k+1}=y_k+hk_1$$

再由此点求得另一斜率

$$k_2=f(t_{k+1},\ \bar{y}_{k+1})=f(t_k+h,\ y_k+hk_1)$$

然后，从 y_k 点开始，既不按该点斜率 k_1 变化，也不按预估点斜率 k_2 变化，而是取两者平均值

$$k=\frac{k_1+k_2}{2}$$

求得校正点 y_{k+1}，即

$$y_{k+1}=y_k+hk$$

该方法可以认为是经改进了的欧拉法。

又将式（2-95）与式（2-83）比较，有

$$\int_{t_k}^{t_{k+1}}f(t,y)\,\mathrm{d}t=\frac{h}{2}(k_1+k_2)=\frac{h}{2}(f_k+f_{k+1})$$

相当于 $[t_k,\ t_{k+1}]$ 区间内的曲边面积被上下底为 f_k 和 f_{k+1}、高为 h 的梯形面积所代替，这样精度提高很多。其几何意义如图 2-33 所示。

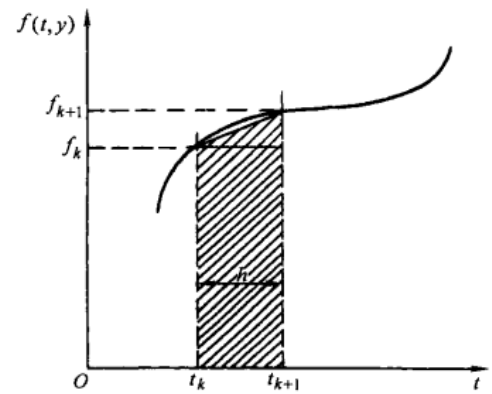


图 2-33　预估-校正法的几何意义

由此而观察 1）、2）、3）法，发现二阶龙格-库塔法的规律是相同的，都是通过 y_k 点先求取斜率 k_1，再以此斜率求取另一斜率 k_2，最后以满足二阶精度为目的，取适当加权系数，求取调整斜率

$$k=b_1k_1+b_2k_2$$

可以说，这也是整个龙格-库塔法的共同规律。于是我们清楚地理解了式（2-92）中求和部分 $\sum_{i=1}^{r}b_ik_i$ 的意义，就相当于一个经多项加权系数 b_i 对多个 y_k 点附近变化斜率 k_i 加以调整的总斜率 k，即

$$k = \sum_{i=1}^{r} b_i k_i \tag{2-96}$$

因而，式（2-92）成为

$$y_{k+1} = y_k + hk$$

它完全是欧拉公式（2-87）的基本形式。但精度随 r 的取值提高为 r 阶精度。于是，按照上述思路不难得到 $r=3$ 时，三阶龙格-库塔公式

$$\begin{cases} y_{k+1} = y_k + \dfrac{h}{4}(k_1 + 3k_3) \\ k_1 = f(t_k, y_k) \\ k_2 = f\left(t_k + \dfrac{h}{3}, y_k + \dfrac{h}{3}k_1\right) \\ k_3 = f\left(t_k + \dfrac{2}{3}h, y_k + \dfrac{2}{3}hk_2\right) \end{cases} \tag{2-97}$$

当 $r=4$ 时，四阶龙格-库塔公式为

$$\begin{cases} y_{k+1} = y_k + \dfrac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) \\ k_1 = f(t_k, y_k) \\ k_2 = f\left(t_k + \dfrac{h}{2}, y_k + \dfrac{h}{2}k_1\right) \\ k_3 = f\left(t_k + \dfrac{h}{2}, y_k + \dfrac{h}{2}k_2\right) \\ k_4 = f(t_k + h, y_k + hk_3) \end{cases} \tag{2-98}$$

对于仿真中遇到的大多数工程实际问题，四阶龙格-库塔法的精度已能满足要求，其截断误差 $o(h^5)$ 与 h^5 同数量级，当步距 h 取得较小时，误差是很小的。

龙格-库塔法无论几阶均属单步法，当然都可以自启动。龙格-库塔公式中的各次斜率 k_i，也称为龙格-库塔系数。

三、关于数值积分法的几点讨论

（一）单步法和多步法

如前所述，欧拉法、龙格-库塔法等，由于计算 t_{k+1} 时刻值 y_{k+1} 只与 t_k 时刻 y_k 有关，故都是单步法，可以自启动。但还有许多算法，由于计算 t_{k+1} 时刻的值 y_{k+1}，要用到 t_k 及过去时刻 t_{k-1}，t_{k-2}，…，t_{k-r} 的值 y_k，y_{k-1}，…，y_{k-r}，于是称为多步法。线性多步法可以表示为以下一般形式

$$y_{k+1} = \alpha_0 y_k + \alpha_1 y_{k-1} + \cdots + \alpha_r y_{k-r} + h(\beta_{-1} f_{k+1} + \beta_0 f_k + \beta_1 f_{k-1} + \cdots + \beta_r f_{k-r}) \tag{2-99}$$

由式中可知，多步法不能从 $t=0$ 自启动，通常需要选用相同阶次精度的单步法来启动，获得所需前 r 步数据后，方可转入相应多步法。

常见多步法有阿达姆斯（Adams）法，其二阶公式为

$$y_{k+1} = y_k + \frac{h}{2}(3f_k - f_{k-1}) \tag{2-100}$$

或

$$y_{k+1} = y_k + hf_{k+1} \tag{2-101}$$

式（2-101）通常也称稳式欧拉法。式中，$f_i = f(t_i, y_i)$（$i = k, k-1, \cdots$）为导函数各时刻值，也就是各时刻的切线斜率。

还有基尔法（Gear），其三阶公式为

$$y_{k+1} = \frac{1}{2}(-3y_k + 6y_{k-1} - y_{k-2} + 6hf_k) \tag{2-102}$$

和

$$y_{k+1} = \frac{1}{11}(18y_k - 9y_{k-1} + 2y_{k-2} + 6hf_{k+1}) \tag{2-103}$$

多步法的特点是在每一时刻上，计算公式简洁，无需求取多个斜率，但无法自启动，需借助其他方法启动。因算式利用信息量大，因而比单步法更精确。

（二）显式和隐式

多步法中，计算 y_{k+1} 时公式右端各项数据均已知，如式（2-100）、式（2-102）类型，则称为显式，在一般表达式（2-99）中，对应 $\beta_{-1} = 0$ 时的情况。若求 y_{k+1} 算式中包含着 f_{k+1}，而 f_{k+1} 的计算又要用到 y_{k+1}，即求解 y_{k+1} 的算式中隐含着 y_{k+1} 本身，如式（2-101）、式（2-103）类型，则称为隐式，对应一般表达式 $\beta_{-1} \neq 0$ 时的情况。

显式易于计算，利用前几步计算结果即可进行递推求解下步结果。而隐式计算需要迭代法，先用另一同阶次显式公式估计出一个初值 $y_{k+1}^{(0)}$，并求得 f_{k+1}，然后再用隐式公式得校正值 $y_{k+1}^{(1)}$，若未达所需要精度要求，则再次迭代求解，直到两次迭代值 $y_{k+1}^{(i)}$，$y_{k+1}^{(i+1)}$ 之间的误差在要求的范围内为止，故隐式算法精度高，对误差有较强的抑制作用，尽管计算过程复杂，造成计算速度慢，但有时基于对精度、数值稳定性等考虑，仍经常被采用，如求解病态（stiff）方程等问题。

（三）数值稳定性

数值积分法求解微分方程，实质上是通过差分方程作为递推公式进行的。在将微分方程差分化的变换过程中，有可能使原来稳定的系统变为不稳定系统。因此，可以说数值积分算法本身从原理上就不可避免地存在着误差，并且在计算机逐点计算时，初始数据的误差、计算过程的舍入误差等都会使误差不断积累，如果这种误差积累能够得到抑制，不会随计算时间增加而无限增大，则可以认为相应的计算方法是数值稳定的，反之则是数值不稳定的。

数值稳定性可以通过对不同数值积分法对应的差分方程的稳定性分析得到。而差分方程的稳定性与采样周期 T（相当于算法公式中的步距 h）有很大关系。所以最简单的数值稳定性判别方法是取两种显著不同的步距进行试算，若所得数据基本相同，则一般是稳定的。当然，这种方法仅适用于简单地估计一下稳定性。详细讨论数值稳定性问题是非常复杂的，这里只给出一种判定算法数值稳定性的常用方法，有兴趣的读者可参阅有关计算方法的文献和专著。

对一般常系数微分方程，其 $f(t_k, y_k)$ 表达形式多种多样，没有办法统一，所以很难得到能适应所有微分方程的数值稳定性判定法。于是，建立一个试验方程

$$\frac{dy}{dt} = f(t,y) = \lambda y, \mathrm{Re}(\lambda) = -1/\tau < 0 \tag{2-104}$$

式中，λ 为试验方程的定常复系数，其实部 $\mathrm{Re}(\lambda)$ 为负，以保证试验方程本身是稳定的，从而才能研究数值算法的稳定性；τ 为系统时间常数，可反映一阶系统的动态性能。

这可以说是常微分方程中最简单的形式，用它来判断一个数值算法是否稳定很能说明问

题。如果一个数值算法连这样简单的方程都不能适应，不能保证其绝对稳定性，求解一般方程也不会稳定。如果能保证其绝对稳定性，虽然不能说求解一般方程也会绝对稳定，但该算法的适应性肯定要好得多。

以欧拉法为例：

$$y_{k+1} = y_k + hf(t_k, y_k)$$

为其递推公式。将试验方程代入，即 $f(t_k, y_k) = \lambda y_k$ 时，有

$$y_{k+1} = y_k + h\lambda y_k = (1 + h\lambda) y_k$$

这是一个一阶差分方程，其特征值为

$$z = 1 + h\lambda$$

要求该方程绝对稳定，必有

$$|z| = |1 + h\lambda| \leqslant 1$$

结合式（2-104），即得到该算法的稳定条件 $h \leqslant 2\tau$，其稳定边界也随之求得为 $|z| = |1 + h\lambda| = 1$，即有 $h = 2\tau$。

显然，算法的稳定与所选步长 h 有关。步长太大，超过稳定条件限制，会造成计算不稳定，因此只允许在一定范围内取值，通常称之为条件稳定格式。从以上推导知，对于显式欧拉法，仿真步长 h 至少应该小于系统时间常数 τ 的两倍。

为使欧拉法能适应一般方程，应更加严格地限制步长 h，取 $h < \tau$，甚至 $h << \tau$，才能保证计算过程的稳定性。

为直观地理解稳定条件意义，往往以 $h\lambda$ 为复平面画出图形，更便于看到 h 与系统时间常数 τ 之间的制约关系对稳定性的影响，如图 2-34 所示。

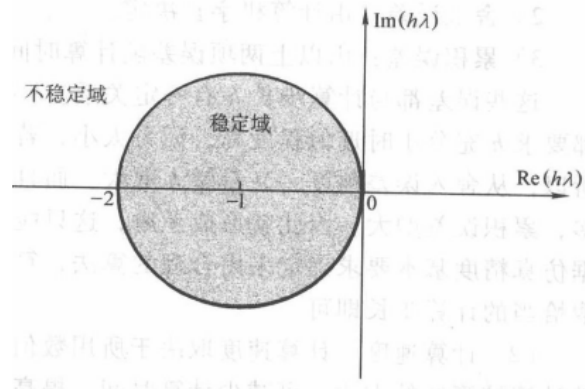


图 2-34　显式欧拉法的稳定域

对其他数值算法，都可仿照上述方法分析其数值稳定性。如已知隐式欧拉法递推公式（2-101）

$$y_{k+1} = y_k + hf(t_{k+1}, y_{k+1})$$

将试验方程代入，得

$$y_{k+1} = y_k + h\lambda y_{k+1}$$

整理，有差分方程

$$y_{k+1} = \frac{1}{1 - \lambda h} y_k$$

其特征值为

$$z = \frac{1}{1 - \lambda h}$$

稳定条件为 $|z| < 1$，但由于 $\mathrm{Re}(\lambda) < 0$，故只要 $h > 0$，因此无论取何值均能满足稳定条件，不受试验方程参数的制约，属于无条件恒稳格式。相应的 $h\lambda$ 复平面稳定域图形如图 2-35 所示。

由图 2-35 与图 2-34 比较可知，不同算法的稳定域有很大差别。

（四）数值算法的选用

MATLAB 语言的控制系统工具箱以及仿真工具 Simulink 中提供了以下几种常用数值算法，可根据实际情况方便地选用。

-Euler 法

-2/3 阶 Runge-Kutta 法

-4/5 阶 Runge-Kutta 法

-Adams 预报-校正法

-Gear 预报-校正法

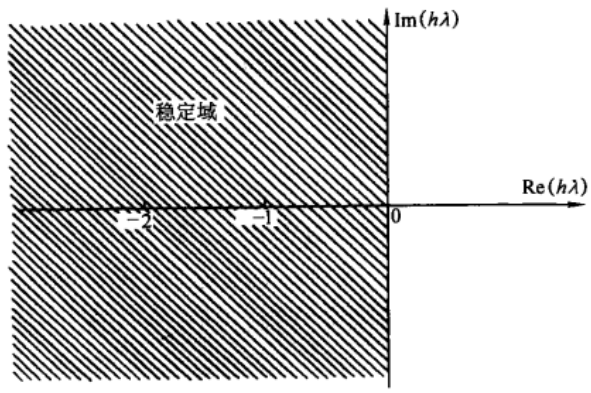


图 2-35 隐式欧拉法的稳定域

作为仿真算法的使用者，不必考虑对各种数值方法的具体编程实现这类基础性、技术性太强的问题，而主要应关心各种方法在使用中会出现的问题，以及如何在仿真过程中恰当地运用这些方法。

一般地说，选用数值方法从以下原则考虑：

（1）精度 仿真结果的精度主要受三项误差影响：

1）截断误差：由算法本身的精度阶次所决定。

2）舍入误差：由计算机字长决定。

3）累积误差：由以上两项误差随计算时间长短累积情况决定。

这些误差都与计算步长 h 有一定关系。h 越小，截断误差就会越小，因为各算法原理上都要求 h 充分小时近似程度高，而 h 太小，若小到计算机字长难以准确表示，则失去意义。所以，从舍入误差角度，又希望 h 取大。而且 h 小，会导致计算步数增加，造成累积次数增多，累积误差增大。因此就形成矛盾，这只能从保证精度前提下，采取折衷办法兼顾。先根据仿真精度基本要求确定采用合理的算法，算法一旦确定，则从控制累积总误差角度考虑，取恰当的计算步长即可。

（2）计算速度 计算速度取决于所用数值方法和计算步长。在满足精度要求的前提下，选计算较简便的方法，可减少计算时间，提高速度。一般可用多步法、显式计算法等计算速度较快的方法。当算法一定，精度只要能得到保证，应尽量选用大步距，同样也可提高速度。

（3）稳定性 数值稳定性主要与计算步长 h 有关，不同的数值方法对 h 都有不同的稳定性限制范围，且与被仿真对象的时间常数也有关。一般所选步长与系统最小时间常数有以下数量级关系

$$h \leqslant (2 \sim 3)\ \tau$$

而多步法、隐式算法有较好的数值稳定性，在对稳定性较注重时，应予以优先选用。

第五节 数值算法中的“病态”问题

一、“病态”常微分方程

在控制系统的分析与设计中，往往会碰到这样的情况，系统方程建立起来后，数值算法

也依照前节所述原则选定了，但求解过程却很不顺利，取不同的计算步长值，得到结果不同，有时差异还很大。

例如，已知系统状态方程

$$\begin{cases}\dot{\boldsymbol{X}} = \boldsymbol{AX} \\ \boldsymbol{X}(0) = (1, 0, -1)^{\mathrm{T}}\end{cases} \tag{2-105}$$

式中

$$X = (x_1, x_2, x_3)^{\mathrm{T}}$$

$$\boldsymbol{A} = \begin{bmatrix} -21 & 19 & -20 \\ 19 & -21 & 20 \\ 40 & -40 & -40 \end{bmatrix}$$

采用四阶龙格-库塔法，取 $h=0.01$，求出 $t=0$ 到 $t=1$ 时刻各状态解如图 2-36a 所示。

由图可见，$t=0.1$s 之后，曲线变化趋于平缓，若仍以 $h=0.01$ 计算，会使得计算时间拖得很长。为加快计算速度，取较大步长 $h=0.04$ 计算。结果成为图 2-36b 所示形式，显然误差很大，所得数据几乎没有参考价值。$h>0.05$ 后，曲线呈发散振荡形式，属于数值不稳定现象，所得数据完全失去意义。

因此，受数值稳定性限制，只能取很小步长仿真运算的系统，若其动态响应与本例情况相似，有长时间的慢变过程，则使得计算速度大受影响。对一些大型系统，仅采用单一步长计算求解，有时会耗费大量运行时间和占用可观的机器容量，甚至使计算无法进行下去。

究其原因，我们发现，例中系统状态方程矩阵 $\boldsymbol{A}$ 的对应特征值差异较大，即满足 $|\lambda \boldsymbol{I} - \boldsymbol{A}| = 0$ 的特征值，$\lambda_i (i = 1,2,3)$ 分别为 $\lambda_1 = -2, \lambda_2 = -40(1+j), \lambda_3 = -40(1-j)$。其中，$|\mathrm{Re}(\lambda_2)| / |\mathrm{Re}(\lambda_1)| > 10$，相差一个数量级。而我们知道，系统特征值 λ_i 在实际系统中反映了动态过渡过程各瞬态分量不同时间常数 τ_i 的作用。$\mathrm{Re}(\lambda_i)$ 的绝对值大，其相应的瞬态分量时间常数 τ_i 小，瞬态过程短暂，系统性能在此时间段内变化剧烈，反之，则变化相对平缓。故而只要计算步长 h 选取不当，就会造成系统性能变化的全貌不能被很好地反映出来，仿真结果

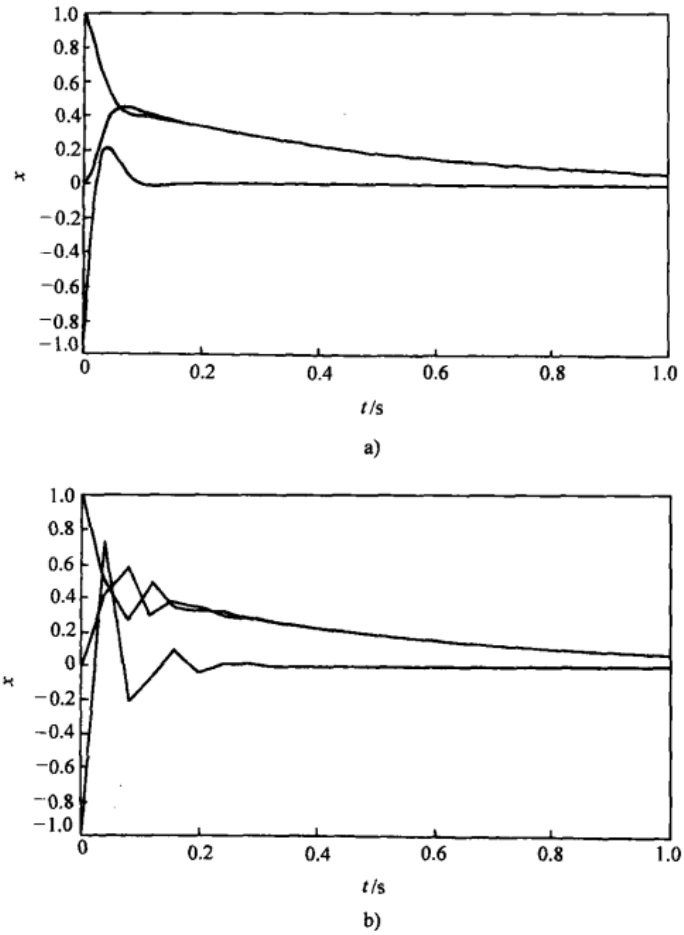


图 2-36　病态系统动态响应的仿真解

a) $h=0.01$　b) $h=0.04$

实际上已经属于数值不稳定的发散情况，失去分析意义。有时会碰到比上例情况更严重的情况，系统特征值实部的绝对值相差不只一个数量级，相差上百倍以上，在仿真中，对步长 h 的选取更为敏感，对算法的数值稳定性要求也更高，而描述这类系统的常微分方程，数学上通常称为“病态（stiff）”方程。表述如下：

一般线性常微分方程组

$$\dot{\boldsymbol{X}}(t) = \boldsymbol{A}\boldsymbol{X}(t) + \boldsymbol{B}\boldsymbol{U}(t), \boldsymbol{X}(t_0) = \boldsymbol{X}_0 \tag{2-106}$$

的系数矩阵 A 的特征值 λ_i 具有如下特征

$$\begin{cases} \mathrm{Re}(\lambda_i) < 0 \\ \max\limits_{1 \leqslant i \leqslant n} |\mathrm{Re}(\lambda_i)| \gg \min\limits_{1 \leqslant i \leqslant n} |\mathrm{Re}(\lambda_i)| \end{cases} \tag{2-107}$$

则称式(2-106)为“病态”方程，用式(2-106)描述的系统则称为病态系统。

更一般地，对非线性常微分方程组

$$\dot{\boldsymbol{X}}(t) = \boldsymbol{F}(\boldsymbol{X}, \boldsymbol{U}, t), \boldsymbol{X}(t_0) = \boldsymbol{X}_0 \tag{2-108}$$

求得其雅可比阵为

$$\mathrm{J} = \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{X}} = \begin{bmatrix} \dfrac{\partial f_1}{\partial x_1} & \dfrac{\partial f_1}{\partial x_2} & \cdots & \dfrac{\partial f_1}{\partial x_n} \\ \dfrac{\partial f_2}{\partial x_1} & \dfrac{\partial f_2}{\partial x_2} & \cdots & \dfrac{\partial f_2}{\partial x_n} \\ \cdots & \cdots & \ddots & \vdots \\ \dfrac{\partial f_n}{\partial x_1} & \dfrac{\partial f_n}{\partial x_2} & \cdots & \dfrac{\partial f_n}{\partial x_n} \end{bmatrix}$$

若 $\boldsymbol{J}$ 阵在 $t = t_0$ 处的特征值 λ_i 也具有式（2-107）特征，则称式（2-108）也为病态方程。同样，该方程所描述的系统也为病态系统。

二、控制系统仿真中的“病态”问题

对一个控制系统而言，其状态方程系数矩阵特征值 λ_i，闭环传递函数分母多项式的根 s_i 或系统闭环极点 p_i 是等价的，当它们之间差异太大，满足式（2-107）关系，该控制系统进行仿真时就被认为是病态系统。

病态系统中绝对值最大的特征值对应于系统动态性能解中瞬态分量衰减最快的部分，它反映了系统的动态响应速度和系统的反应灵敏程度，一般与系统中具有最小时间常数 $T_{\min}$ 的环节参数有关，如系统中的控制器、反馈元件等要求反应灵敏的环节和参数。

而病态系统中绝对值最小的特征值对应于瞬态分量衰减最慢的部分，它决定了整个系统的动态过渡过程时间的长短，一般与系统中具有最大时间常数 $T_{\max}$ 的环节参数有关，如系统中具有较大惯性的控制对象（温度、压力、流量等）各环节和参数。

对这类病态系统的数值求解，计算中往往存在很大困难。为反映出系统的灵敏程度，对其变化最剧烈的部分给出准确的描述，则要求计算步长 h 取得很小，否则将丢失有用的数据信息，造成数值不稳定，其数据结果没有参考价值，如前节例子。但这样做带来的问题是，当系统性能变化相对平缓时，步长太小，使得计算速度慢，数值变化幅度小，要达到要求的

求解时间，耗费时间长，工作效率低，并且 h 取得太小，受计算机字长限制，会引起舍入误差增大，计算时间越长，引起总的累积误差越大，导致计算失败。

由于系统总体动态响应时间是由最小特征值（或相应最大时间常数）决定的，因此为节省时间，会取较大计算步长 h，而计算步长取值过大造成的问题不仅仅是数值不准确，严重情况下，会出现数值不稳定的发散现象，计算结果更是毫无价值可言。

所以，对病态系统的仿真，需寻求更合理的数值算法，以解决病态系统带来的选取计算步长与计算精度、计算时间之间的矛盾。

三、“病态”系统的仿真方法

通过以上分析，可知病态系统的求解，需要采用稳定性好、计算精度高的数值算法，并且允许计算步长能根据系统性能动态变化的情况在一定范围内作相应变化，即采用自动变步长数值积分法。

解决这类问题的有效数值算法有多种，这里重点介绍一种隐式吉尔（Gear）法。

由本章第四节知，多步法、隐式算法精度高，对误差有较强抑制作用，故数值稳定性好，尽管计算过程复杂，从求解病态方程角度考虑，却是一种十分有效的实用算法。隐式吉尔法即为符合要求的算法之一。

将式（2-99）重写如下

$$y_{k+1}=\alpha_0 y_k+\alpha_1 y_{k-1}+\cdots+\alpha_r y_{k-r}+h\left(\beta_{-1}f_{k+1}+\beta_0 f_k+\cdots+\beta_r f_{k-r}\right)$$

吉尔已证明：令系数 $\beta_0=\beta_1=\cdots=\beta_r=0$，并取系数 α_i（$i=0$，1，…，r），以及 β_{-1} 如表 2-3 形式时，此递推公式对病态方程求解计算过程是数值稳定的。r 表示所取隐式吉尔法的精度阶次。

表 2-3　隐式吉尔法系数表

r	α_0	α_1	α_2	α_3	α_4	α_5	β_{-1}
1	1	0	0	0	0	0	1
2	4/3	-1/3	0	0	0	0	2/3
3	18/11	-9/11	2/11	0	0	0	6/11
4	48/25	-36/25	16/25	-3/25	0	0	12/25
5	300/137	-300/137	200/137	-75/137	12/137	0	60/137
6	360/147	-454/147	400/147	-225/147	72/147	-10/147	60/147

第四节中式（2-103）即为具有三阶精度的三步隐式吉尔法递推公式。表 2-3 中取 $r=1$ 时，递推公式为

$$y_{k+1}=y_k+hf_{k+1}$$

它是一个隐式公式，形式却与最简单的欧拉法相同，不同的是以 $(y_{k+1}-y_k)/h$ 近似代替 y 在 $t=t_{k+1}$ 点的导数 $f(t_{k+1},\ y_{k+1})$ 即 f_{k+1}，故又称后退欧拉公式。根据数值稳定性分析可知其为恒稳格式，即步长 h 的取值，理论上讲，将不影响数值计算的精度，保证数据结果是收敛于实际值的，这正是隐式吉尔法的突出特点。通过自动改变步长 h，适应系统不同特征值 λ_i 下相应的动态变化性能，从而为解决病态问题提供了理论依据。

隐式吉尔法虽然从理论上说十分适应于病态系统仿真，但因其为隐式多步法，实用时需

解决好自启动、预估迭代和变阶变步长问题。

(1) 自启动 r 阶多步算式无法自启动，需要用单步法求得前 r 步值，然后才能转入多步法连续计算下去。因此采用单步法求得的 r 个出发值必须保证所求数值的精度，否则即使取得前 r 步值，但由于精度太差会造成后面计算无法正常进行。

(2) 预估迭代 隐式法不能直接求得 y_{k+1}，又需要用显式法做预估，然后通过有效迭代，求得所需值。因此，迭代方法要求收敛性良好，否则虽然理论上隐式吉尔法稳定域很宽，但迭代过程若对步长 h 敏感，也会在大步长时造成计算数值发散，得不到正确结果。

(3) 变步长 对病态系统仿真总是要求在计算过程中采用变步长，即系统变量初始阶段变化剧烈，要求用小步长细致描述刻划，而后随着变化趋缓，可逐步放大步长，以减少计算时间。

此外，对不同精度要求的系统仿真，要考虑变阶次问题。即为减小每一步计算的截断误差，以提高精度时，应选较高阶次，而当精度要求较低时，为减少工作量，则应选取较低阶次。

按以上考虑，实际计算中首先应以减小步长来满足精度要求，当达到最小步长还未满足精度要求，则应提高该方法的阶次；当在 r 阶方法连续若干步的计算中精度始终满足要求时，则可以适当降低为 $r-1$ 阶方法继续运算。

另外，是否满足精度要求可通过对所求各时刻数值解的相互关系和所用方法列出误差估计公式来加以判断。每计算一步，将估计误差 ε 与事先规定的误差精度 ε_0 相比较，当 $\varepsilon \leqslant \varepsilon_0$，则结果有效，转入下一步计算；当 $\varepsilon > \varepsilon_0$，则本步结果无效，改变步长或阶次后重新计算本步，直至符合要求；若 $\varepsilon \ll \varepsilon_0$，则说明计算过程还可大大加快，增大步长或降低阶次后转入下一步计算。

关于以上几点，许多文献上都给出了相应的实用算法和实现策略，欲深入了解和学习的读者请参阅书后有关参考文献。

第六节 数字仿真中的“代数环”问题

一、问题的提出

反馈是控制系统中普遍存在的环节，在进行数字仿真时，计算机按照一定的时序执行相应的计算步骤，对于反馈回路就有一个输入和输出计算顺序的问题。在相当普遍的条件下，当一个系统的输入直接取决于输出，同时输出也直接取决于输入时，仿真模型中便出现“代数环”问题[77-79]。

下面结合一个实例给出“代数环”的定义。图 2-37 所示为一个最简单的“代数环”问题。仿真模型的输出反馈信号作为输入信号的一部分，在进行仿真时，按正常的计算顺序应该先计算模块的输入，然后再计算由输入驱动的输出。然而，由于输入与输出相互制约，这就形成了一个死锁环路，也就是所谓的“代数环”问题。当一个仿真模型中存在一个闭合回路，并且回路中每个模块/环节都是直通的，即模块/环节输入中的一部分直接到达输出，这样一个闭合回路就是“代数环”。

显而易见，“代数环”问题会严重影响仿真速度，某些情况下还会降低控制系统仿真的

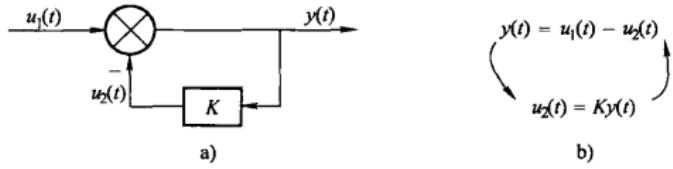


图 2-37　“代数环”问题简例

a）仿真模型　b）死锁环路

精度，甚至导致仿真系统停滞。对于存在“代数环”的反馈系统模型，需要仿真者事先采取必要措施加以避免。

对于简单的“代数环”问题，MATLAB/Simulink 系统仿真软件提供了“牛顿迭代算法”对“代数环”问题进行求解。“牛顿迭代算法”是一种基于一阶 Taylor 级数展开的逐次迭代逼近法，在迭代算法的每一步都需要进行多次求导运算。因此，随着模块功能复杂程度的加大，精确度要求的提高，迭代计算量将会大幅增加，从而导致运行速度剧降，仿真效率很低。此外，“牛顿迭代算法”具有一定的收敛条件，当收敛条件不满足时，Simulink 对“代数环”的求解误差较大。

随着控制系统模型复杂性的增加和非线性环节的影响，“代数环”问题变得非常隐蔽而难以为人所认识。为保证系统仿真的速度和精确度，高效地应用 MATLAB/Simulink 等仿真软件，有必要了解“代数环”问题产生的条件和有效的消除方法。

二、“代数环”产生的条件

如前所述，“代数环”是一种反馈回路，但并非所有的反馈回路都是“代数环”，其存在的充分必要条件是：在系统仿真模型中，存在一个闭合路径，该闭合路径中的每一个模块/环节都是直通模块/环节。

所谓直通，指的是模块/环节输入中的一部分直接到达输出。如果一个反馈回路的正向通道和反向通道都由直通模块组成，则此反馈回路一定构成“代数环”。对于复杂反馈回路来说，只要能够找到由直通模块构成的闭合路径，则也一定构成“代数环”。

常见的几种代数环现象如图 2-38 所示。

在应用 MATLAB/Simulink 进行图形化建模的时候，应该对其模块库中哪些模块及其在什么条件下有直通特性有所了解，从而可以预见到“代数环”的存在，也可以为消除“代数环”以及避免产生新的“代数环”提供帮助。表 2-4 列出了 MATLAB/Simulink 模块库中一些典型的直通模块[77]。

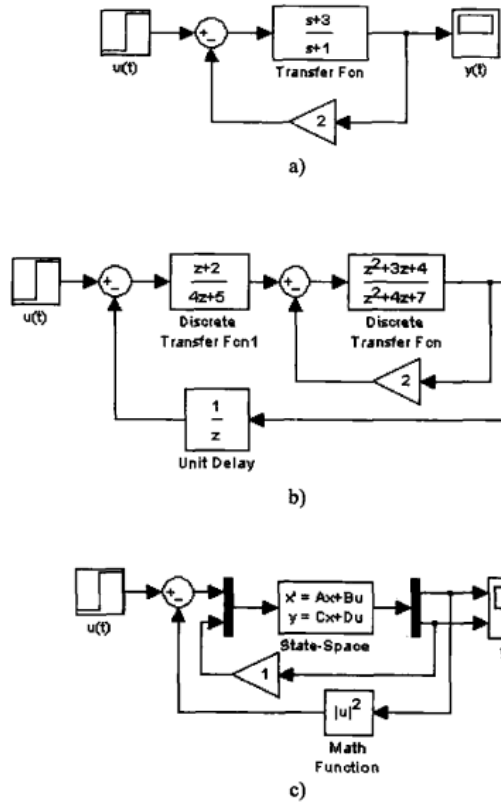
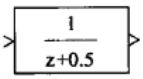
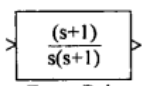
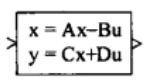
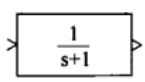


图 2-38　几种常见的代数环现象

a）“代数环”Ⅰ　b）“代数环”Ⅱ　c）“代数环”Ⅲ

表 2-4 Simulink 模块库中一些典型的直通模块

模块	所属模块库	说 明
	Simulink/Math	任意情况
	Simulink/Discrete	当离散传递函数的分子与分母阶数相同时
	Simulink/Continuous	从初始条件输入端到输出的直通
	Simulink/Math	任意情况
	Simulink/Math	任意情况
	Simulink/Continuous	当极点数目与零点数目相同时
	Simulink/Continuous	当 D 矩阵非 0 时
	Simulink/Continuous	当传递函数的分子与分母阶数相同时

在 MATLAB/Simulink 的仿真模型中，产生“代数环”的一般条件可归纳如下，在具体仿真实验中要予以充分注意。

1）前馈通道中含有信号的“直通”模块，如比例环节或含有初值输出的积分器。

2）系统中的大部分模型表现为非线性。

3）前馈通道传递函数的分子与分母同阶。

4）用状态空间描述系统时，输出方程中 D 矩阵非 0。

三、消除“代数环”的方法

“代数环”的表现形式是多种多样的，消除“代数环”的方法也不尽相同，一般可以分为两类，一类为变换法，另一类为拆解法[77,80]。

（一）变换法

“代数环”在形式上是一种数字仿真模型，对应的是数学模型，通常表现为一个方程或方程组。当方程的右边项中包含有方程的左边项时，如果用 Simulink 去直接实现该方程，将产生“代数环”。如果先将原始数学模型进行变换，使得方程的右边项中不包含有方程的左边项，然后再用 Simulink 去实现，则可以消除“代数环”，其本质上是将“代数环”隐函数变换为显函数的方法。

例如，设一个系统的描述方程为

$$\begin{cases} \dot{x} = u - 0.5y \\ y = 4x + 2\dot{x} \end{cases} \tag{2-109}$$

式中，u，x，y 分别代表输入量、状态变量和输出量。该系统直接利用 Simulink 建模，如图 2-39 所示。

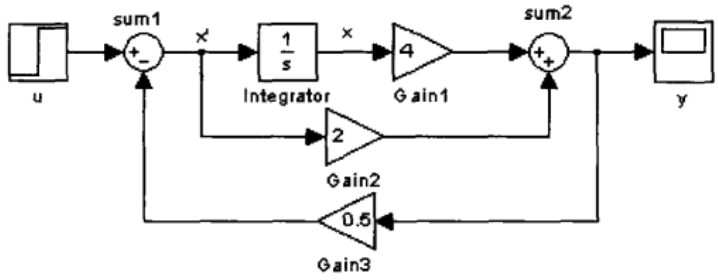


图 2-39　Simulink 仿真模型

仿真过程中 MATLAB 提示如下：

Warning: Block contains 1 algebraic loop (s).

Found algebraic loop containing block (s):

'Gain3','Sum1','Gain2','Sum2'.

它说明该模型在仿真时含有“代数环”，是由于’ Sum1’,’ Gain2’,’ Sum2’,’ Gain3’四个直通模块构成了一个闭合回路。我们可以采用变换法消除代数环，系统方程明显为隐函数，通过对方程各变量进行代换调整，可以得到如下新的系统方程

$$\begin{cases} \dot{x} = -x + 0.5u \\ y = 2x + u \end{cases} \tag{2-110}$$

重新建立 Simulink 仿真模型，如图 2-40 所示。

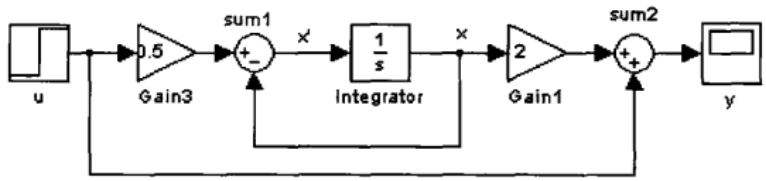


图 2-40　Simulink 仿真模型

虽然仿真模型中依然存在反馈回路，但其中前向通道的积分环节不是直通模块，因此该反馈回路也就不再构成“代数环”。

然而，用变换法消除“代数环”存在两个方面的限制。首先，并不是所有的隐函数都

可以求解得到显函数；其次，原始的数学模型往往反映了仿真对象的物理结构，按照物理结构构造仿真模型可以实现所谓的同构仿真，按照变换后得到的数学模型构造的仿真模型则只能实现同态仿真，同构仿真比同态仿真具有更好的可信度，也具有更大的灵活性。

一般情况下，为避免“代数环”的产生，在系统建模与列写仿真系统的微分方程时，习惯上“将最高次微分项全部列于方程左边，其他阶次微分项及输入与干扰等项列于方程右边”。

（二）拆解法

用拆解法消除“代数环”基于这样一个认识：“代数环”是一个闭合回路，而且回路中的每一个模块都必须是直通模块，在保持功能不变的同时，如果能够在回路中产生一个非直通模块，则该“代数环”就被拆解了。“代数环”的拆解有多种方法，受篇幅所限，这里仅介绍其中的三种。

（1）插入存储器模块拆解“代数环” 存储器模块是 Simulink 库中的一个模块，其功能是将当前的输入采样保持一个时间步，然后再输出。存储器模块在每一个时间步的输出都是其在上一个时间步的输入，因此存储器模块是一个非直通模块。如图 2-41 所示，将存储器模块插入代数环中可以拆解“代数环”。当然，引入存储器模块肯定对原系统的精度有影响。特别是对相位稳定裕量不大的系统，有可能产生振荡，因为存储器模块实际上是一个延迟环节。

（2）通过模型替代或重构的方法消除直通模块 在一定的条件下，“代数环”一些直通模块可以用具有相同功能的非直通模块替代。从系统原理出发，也可以通过重构部分模型，从而消除直通模块，实现“代数环”的拆解。例如，计数器模块用于对输入脉冲计数，是一个直通模块。为了避免产生“代数环”问题，可以用如下离散函数取代计数器模块

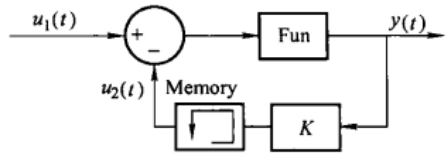


图 2-41 插入存储器模块拆解“代数环”

$$\begin{cases} \dfrac{Y(z)}{U(z)} = \dfrac{1}{z-1} \\ y(n+1) = y(n) + u(n) \end{cases} \tag{2-111}$$

式中，$Y(z)$ 和 $U(z)$ 分别是输出变量 $y(n)$ 和输入变量 $u(n)$ 的 z 变换。可见，如果将该离散传递函数模块的输入信号幅度量化为单位值，同时适当设置模块的采样时间，则该模块可以准确地实现计数器的模块的功能。然而该模块是非直通模块，可以有效消除“代数环”问题。然而，当控制系统的结构比较复杂或模型重构较困难时，该方法的实现有一定的难度。

（3）用 Simulink 提供的专门手段拆解代数环 Simulink 提供了一些专门手段来拆解“代数环”，例如代数约束模块、积分模块的状态输出端等，这些手段可以解决一些特定的“代数环”问题。例如，从积分模块的输入端口到输出端口是非直通的，但从积分模块的初始值输入端口到输出端口，以及从复位输入端口到输出端口却都是直通的。因此，如果从积分模块的输出端口引出的信号再经过一些直通模块后又反馈到积分模块的初始值输入端口或者复位输入端口，则构成一个“代数环”。为了解决这个问题，Simulink 专门为积分模块设计了一个状态端口，其输出与输出端口完全相同，仅在内部计算的时序上有细微区别，而无论是从积分模块的初始值输入端口还是从复位输入端口到状态端口都是非直通的。因此，当出

现上述的“代数环”问题时，可以从积分模块的状态端口引出信号，从而“代数环”就被拆解了。

关于“代数环”的问题，参考文献[77，80]也给出了相应的实用算法和实现策略，感兴趣的读者不妨参阅之，并作进一步的探讨。

第七节　问题与探究——电力电子器件建模问题

一、问题提出

在本章第二节的“三相电压型 PWM 整流器系统”建模中，所涉及到的电力电子器件（如 MOSFET、IGBT）都是按理想的开关模型（“0”、“1”状态）处理的；然而，当我们在研究微观时间尺度下电力电子器件（电压、电流）动态响应（或器件特性）的时候，就必须对电力电子器件建立更为精确的仿真模型，以利于电力电子系统的动态仿真（全时间范围）与实际系统的全状态模拟，其原理如图 2-42 所示。

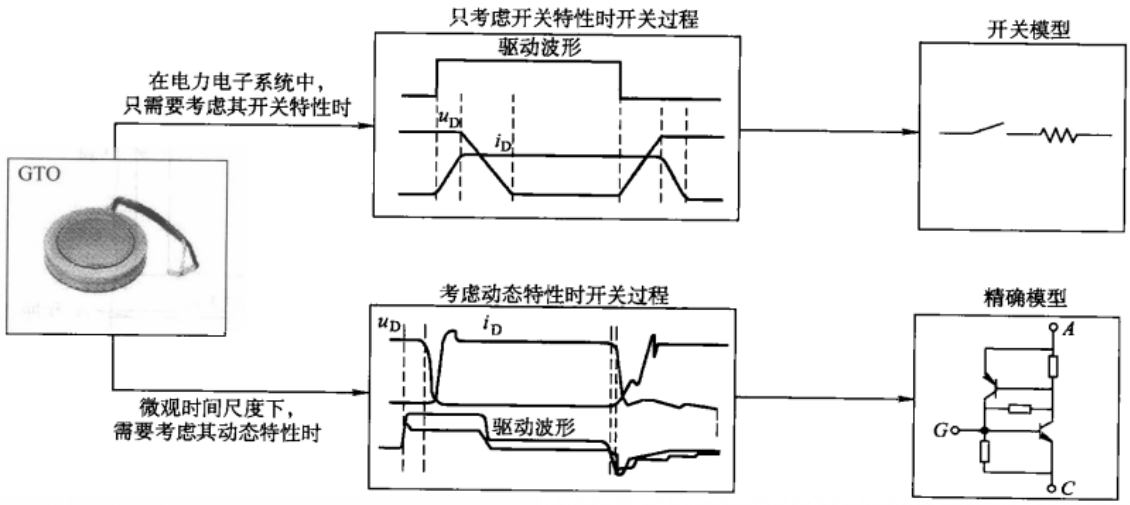


图 2-42　仿真模型等效原理图

显然，在研究电路系统特性或者器件本身特性时，当需要在微秒（μs）尺度里研究器件的动态特性时，电力电子器件的模型不能沿用“0”、“1”状态模型；同时，其动态模型也难以用微分方程或传递函数的数学形式来描述。这是因为电力电子器件实际工作过程及其各种特性和它的半导体结构密切相关，难以建立电力电子器件动态过程的数学描述，需要考虑另外的建模方式。

电力电子器件的精确模型主要应用在：器件模型换向过程（微观时间尺度上）、元器件张力、功率消耗、设计器件缓冲电路等情况下的动态分析。

二、建模机理

（一）电力电子器件建模需考虑的问题

对于功率半导体器件模型的发展，除了考虑半导体器件在建模时所考虑的一般问题和因

素之外，在建立比较精确的仿真模型时，以下几个问题必须优先考虑，这些问题在低功率器件中不成问题，但在功率电子器件中这几个问题支配了器件的静态和动态特性。

（1）电阻系数的调制　为了承受较高的电压，功率半导体器件一般都有一定厚度的掺杂半导体层，当器件导通时，这个层决定导通压降和功率损失。这个电阻随电压和电流变化而变化，具有非线性电阻的特性。

（2）电荷存储量　对于双极型器件而言，当处于导通状态时，载流子电荷被存储在低掺杂区域，这些载流子电荷在器件阻断之前，必须尽快地被移走，该过程是引起开关延时和开关损耗的根本原因。我们以往使用的用于仿真的器件模型都是一种准静态模型，这就意味着电荷的分布是器件两端瞬时电压的函数，这根本不适于功率电子器件，若要准确描述器件的动态特性，就需要导入器件的基本物理方程。在暂态瞬间，功率器件的低掺杂区域内的载流子电荷随着时间和位置进行变化。如图 2-43 所示，IGBT 单元细胞物理结构就可以等效出很多非线性电容。

（3）MOS 电容　MOSFET、IGBT、MCT 等控制门极通常都是绝缘门极，这样的门极都具有比较大的门极电容，这个电容受门极电压的影响，是个非线性电容。影响最大的是门极和阳极之间的电容，在应用电路中门极是输入端，而阳极通常是输出端，通过两极电容形成的反馈作用，对开关的特性产生了较大的影响。

（4）电热交互作用　由于功率损耗，电力电子器件在工作时产生大量的热能，器件的特性同器件的温度有极大的关联，因此变化的温度对器件的特性产生了影响，考虑到热能对器件特性的影响，也需要对器件的电热效应建模。

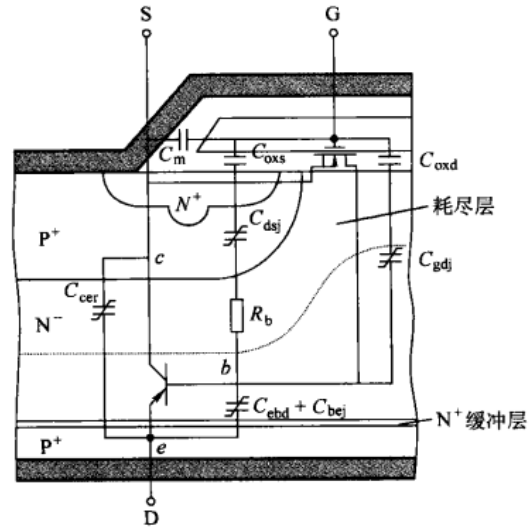


图 2-43　IGBT 单元细胞物理结构及电容分布

（5）击穿　击穿不仅仅发生在器件损坏时，有时正常的应用也会发生击穿现象。功率电子器件的击穿大多数是雪崩击穿，但有时也有齐纳击穿发生。

（二）电力电子器件的建模方法

电力电子器件有各种各样电路模型，所以也存在许多建模方法。目前使用的建模方法主要有数值建模法和子电路建模法。

（1）数值建模法　数值模型一般是直接利用半导体功率器件的物理方程求解而得到模拟结果的一种建模方法。通过这种建模方法得到的电力电子器件数值模型可以称为数值模型（或者微模型），其模型参数与物理原理密切相关，这类模型的特性比较接近器件的实际特性，但是它的参数比较复杂，用户使用起来很不方便，对计算机的要求也非常高。

（2）子电路建模法　这种模型一般都建立在已有的通用电路仿真平台上（如 Saber、Pspice），根据需要建模的电力电子器件特性，利用仿真平台器件库中已有的器件，搭出满足电力电子器件静态和动态特性的模型来。当然这种子电路模型可以很简单，也可能很复杂。子电路型电力电子器件模型的仿真精度一方面取决于模型本身结构，还取决于仿真平台的计算精度和仿真平台中已有的模型精度。所以子电路型电力电子器件模型的精度不会十分

理想，而且由于模型的结构有时过于复杂（有时为了追求仿真精度），仿真时要花费许多时间。但它的优点是物理概念清晰，建模过程简单，容易理解和便于掌握。

（三）电力电子器件的模型参数和模型参数的提取

一个模型是否精确不仅仅取决于模型本身，还取决于模型参数。实际上用于电路仿真的器件模型是否对电路的设计者有价值，取决于模型的参数系列是否可靠。

简单的模型只有较少的参数，这些参数来源于对观测到器件工作特性曲线的拟合。而精确的模型则需要大量的模型参数，因为这些模型是根据器件的物理特性和器件结构建模的，具体请参照参考文献[81-84]。

（四）电力电子器件的子电路模型建立的一般过程（以功率二极管为例）

（1）基本结构以及工作原理分析　如图 2-44 所示，功率二极管为 PN 结构，但由于掺杂工艺和浓度不同其空间电荷分布也有所不同。

（2）工作特性分析　工作特性包括静态工作特性和动态工作特性。这是器件的外特性，可以通过实验观测到，是建立器件模型的重要依据。

图 2-45 所示为功率二极管的静态特性曲线。对于功率二极管的静态特性我们已经很了解，这里不再详述。

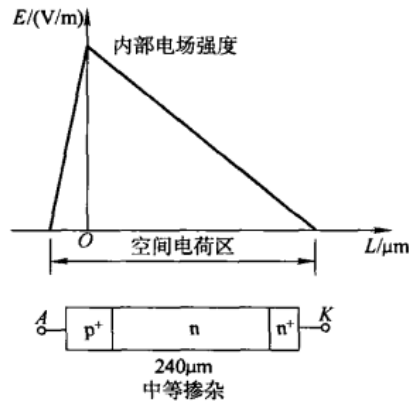


图 2-44　功率二极管结构图

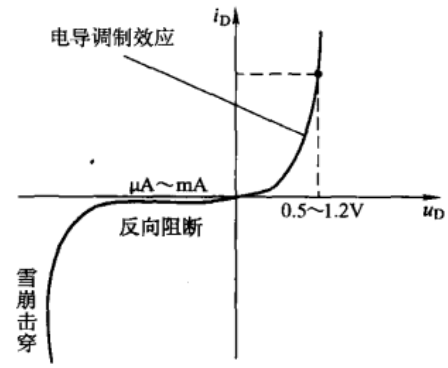
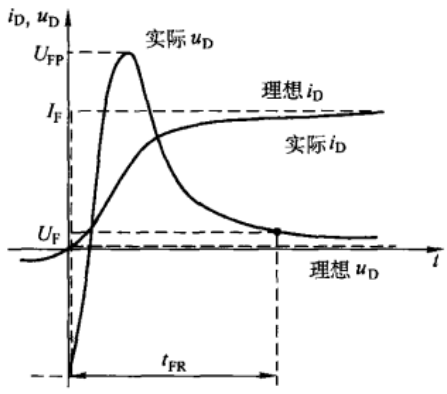
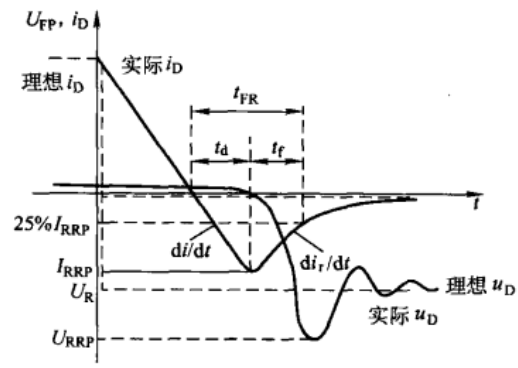


图 2-45　功率二极管静态特性曲线



a)



b)

图 2-46　功率二极管的动态特性

a）开通特性　b）关断特性

如图 2-46 所示，功率二极管的动态特性包括开通和关断特性（当然它的动态特性和其物理结构密切相关）。图中将实际电压电流和理想状态下的响应曲线作了比较，可见实际过程中功率二极管受其内部结构影响与理想状态下响应有着很大差别。

（3）等效物理结构的确定　根据需要建模的电力电子器件特性，对器件的半导体物理结构进行精确地分析得到器件精确的物理等效结构，并通过建立相应的物理方程来得到相应等效元件的参数表达式。

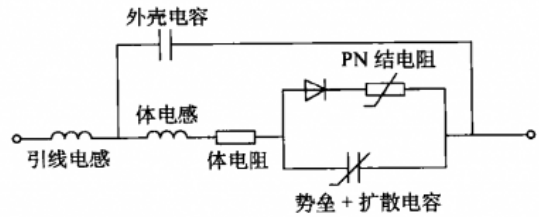


图 2-47　功率二极管的等效物理结构

由于功率二极管结构简单，根据上述方法得到其等效物理结构如图 2-47 所示。

（4）参数提取　根据具体器件的实际特性曲线以及实验测得数据可对器件各等效器件赋值，最终得到完整的仿真模型，可用于仿真实验。

三、问题探究

下面几个问题，感兴趣的读者可以根据相关参考文献探究之。

1）试基于 Pspice 工具软件平台建立如图 2-48 所示 IGBT 的子电路模型，相关内容可参照参考文献[85-87]。

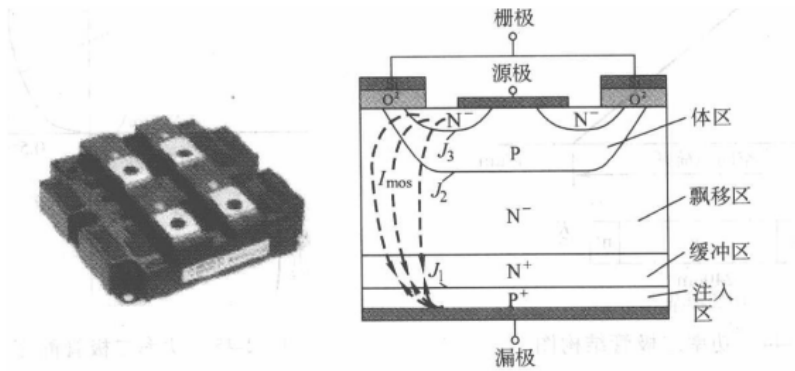


图 2-48　IGBT 及其结构剖面图

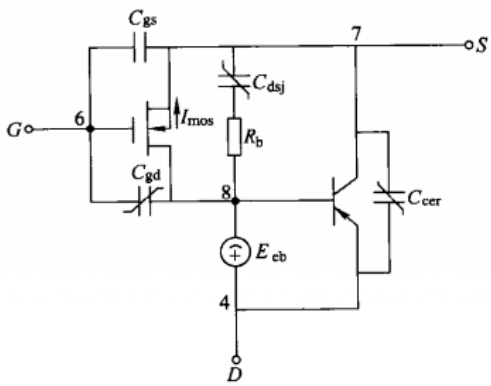


图 2-49　IRGBC40S 的子电路原理模型

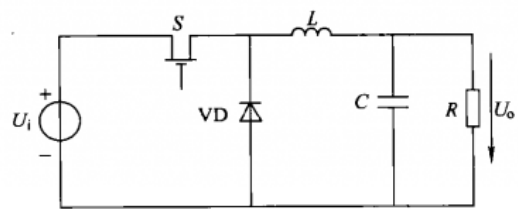


图 2-50　BUCK 电路原理模型

2）参考文献[81]给出了美国 IR 公司的产品 IRGBC40S 型 IGBT 的仿真模型（图 2-49），根据参考文献[81]中的参数，应用 Pspice 工具软件对其进行仿真实验，并根据实验结果说明其动态特性的有效性与正确性。

3）针对图 2-50 所示的 BUCK 电路原理模型，试比较使用开关模型的仿真实验、使用子电路模型的仿真实验与实物实验三者结果的差异，并给出相应结论。

小　结

本章主要讲述控制系统数字仿真的基础内容，现将主要内容归纳如下：

1）控制系统的数学模型是对系统进行计算机仿真的基础。本章介绍了线性系统微分方程、状态方程、传递函数、零极点增益和部分分式等数学模型的表示方法和相应的模型参数表示方法。

2）为使所建立的模型方便地应用 MATLAB 语言进行处理，模型参数采用 MATLAB 语言控制系统工具箱中相应的格式，即

微分方程、传递函数：（*num*，*den*）；

状态方程：　　（***A***，***B***，***C***，***D***）；

零极点增益：　　（***Z***，***P***，*K*）；

部分分式：　　（*R*，*P*，*H*）。

3）数学模型各种形式是为适应不同的分析与设计要求而建立的，它们之间均可以通过一定方法相互转换。MATLAB 语言为此提供了方便可靠的数学模型转换函数 ss2tf（）、tf2ss（）、tf2zp（）等。

4）控制系统建模一般采用机理法、统计法和混合法，又称之为一次模型化。需根据对系统内部结构、特性或是外部输入、输出数据的了解和掌握程度，确定采用何种建模方法更能准确反映系统中各物理量变化规律的动力学特性。系统建模对最终的数字仿真结果有直接影响，应予以充分重视。

5）实现问题就是根据控制系统的传递函数描述求取其相应的状态空间描述；计算机仿真技术的实现问题更具体，就是要将一次模型化得到的系统数学模型，再加以二次模型化，得到可在数字计算机上运行求解的仿真模型。

6）数值积分法是计算机求解一阶微分方程的有效手段。欧拉法最简单易行，且是其他各种数值积分算法的基础，但其截断误差大，不能满足一般工程的精度要求；龙格-库塔法是控制系统仿真最常用的算法，可以根据对仿真精度的不同的要求，选取相应的阶次，一般情况下采用三阶或四阶龙格-库塔法已能满足较高精度的需要。数值积分的单步与多步、显示与隐式等各种方法各有特点，应根据系统仿真的需要灵活应用。

7）数值稳定性是指数值计算过程中，各种误差等积累能否得到很好抑制，是否不会随计算时间增加而不断增大，所得的数值结果是否逼近实际结果等。通过试验方程式（2-104）可以对数值积分方法的数值稳定性作出判断，大致估计出不同方法对计算步长 h 的限制范围。对 h 无限制的方法，称无条件恒稳格式；对 h 有限制的方法，则称条件稳定格式。

8）选用数值算法应从精度、计算速度和稳定性三方面要求来综合考虑。首先保证算法的数值稳定性，其次是满足精度要求，然后再尽可能提高计算速度，减少计算步骤和计算时

间。

9）“病态”系统由于其系统特征值之间的实部绝对值相差过大，造成对所用算法数值稳定性要求很高，对步长的选取非常敏感，故求解“病态”系统需采取稳定性好、精度高，能自动变步长的数值积分法。隐式吉尔法是具有以上特点的常用算法之一。

10）“病态”问题与“代数环”问题是数字仿真中经常遇见而不为人们所重视的问题，往往给仿真结果带来麻烦，需要我们在理解原理的基础上予以注意。

11）以微分方程为基础的“数学模型”是系统模型的主要形式。但是，有些工况难以实现准确的“数学建模”（如电力电子器件的数学建模、水轮发电机组系统的建模等问题）。因此，往往采取“中间变换”的手段处理（如电力电子器件建模的中间变换就是基于 *LRC* 等效的模型）。

习　题

2-1　思考题：

1）数学模型的微分方程、状态方程、传递函数、零极点增益和部分分式五种形式，各自有什么特点？

2）数学模型各种形式之间为什么要相互转换？

3）控制系统建模的基本方法有哪些，它们的区别和特点是什么？

4）控制系统计算机仿真中的“实现问题”是什么含义？

5）数值积分法的选用应遵循哪几条原则？

2-2　用 MATLAB 语言求下列系统的状态方程、传递函数、零极点增益和部分分式形式的模型参数，并分别写出其相应的数学模型表达式。

1）
$$G(s)=\frac{s^3+7s^2+24s+24}{s^4+10s^3+35s^2+50s+24}$$

2）
$$\dot{\boldsymbol{X}}=\begin{bmatrix}2.25 & -5 & -1.25 & -0.5\\ 2.25 & -4.25 & -1.25 & -0.25\\ 0.25 & -0.5 & -1.25 & -1\\ 1.25 & -1.75 & -0.25 & -0.75\end{bmatrix}\boldsymbol{X}+\begin{bmatrix}4\\2\\2\\0\end{bmatrix}u$$

$$y=[0\quad 2\quad 0\quad 2]\,\boldsymbol{X}$$

2-3　用欧拉法求下面系统的输出响应 $y(t)$ 在 $0\leqslant t\leqslant 1$ 上，$h=0.1$ 时的数值解。

$$y'=-y,\ y(0)=1$$

要求保留 4 位小数，并将结果与真解 $y(t)=e^{-t}$ 比较。

2-4　用二阶龙格-库塔梯形法求解题 2-3 的数值解，并与欧拉法求得结果比较。

2-5　用四阶龙格-库塔法求题 2-3 数值解，并与前两题结果比较。

2-6　已知二阶系统状态方程为

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix}=\begin{bmatrix}a_{11} & a_{12}\\ a_{21} & a_{22}\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}+\begin{bmatrix}b_1\\ b_2\end{bmatrix}u;\quad\begin{bmatrix}x_1(0)\\ x_2(0)\end{bmatrix}=\begin{bmatrix}x_{10}\\ x_{20}\end{bmatrix}$$

写出取计算步长为 h 时，该系统状态变量 $\boldsymbol{X}=[x_1,\ x_2]$ 的四阶龙格-库塔法递推关系式。

2-7　单位反馈系统的开环传递函数如下

$$G(s)=\frac{5s+100}{s(s+4.6)(s^2+3.4s+16.35)}$$

用 MATLAB 语句、函数求取系统闭环零极点，并求取系统闭环状态方程的可控标准型实现。

2-8　用 MATLAB 语言编制单变量系统三阶龙格-库塔法求解程序，程序入口要求能接收状态方程各系

数阵（$\boldsymbol{A}$，$\boldsymbol{B}$，$\boldsymbol{C}$，$\boldsymbol{D}$），和输入阶跃函数 $r(t) = R \cdot 1(t)$；程序出口应给出输出量 $y(t)$ 的动态响应数值解序列 y_0，y_1，…，y_N。

2-9　用题 2-8 仿真程序求解题 2-7 系统的闭环输出响应 $y(t)$。

2-10　用式（2-95）梯形法求解试验方程 $y' = -\dfrac{1}{\tau}y$，分析对计算步长 h 有何限制，说明 h 对数值稳定性的影响。

2-11　图 2-51 所示为斜梁-滚球系统原理，若要研究滚球在梁上位置的可控性，需首先建立其数学模型。已知力矩电动机的输出转矩 M 与其电流 i 成正比（即 $M = ki$），横梁为均匀可自平衡梁（即当电动机不通电且无滚球时，横梁可处于 $\theta = 0$ 的水平状态），试建立系统的数学模型，并给出简化后系统的动态结构图。

（提示：建立微分方程→简化→求 $\dfrac{X(s)}{I(s)}$）

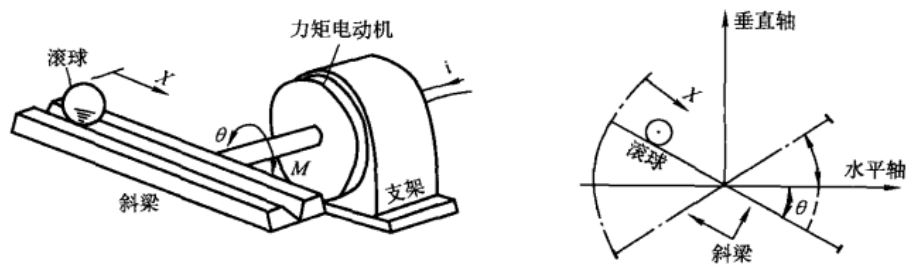


图 2-51　题 2-11“斜梁-滚球”系统原理图

2-12　如图 2-52 所示双水箱系统中，q_{in} 为流入水箱 1 的液体流量，q_{out} 为流出水箱 2 的液体流量，试依据液容与液阻的概念，建立 $Q_{out}(s) \propto [Q_{in}(s), H_1(s), Q_1(s), H_2(s)]$ 的系统动态结构图。

$$\left(\text{提示：}\frac{Q_{out}(s)}{Q_{in}(s)} = \frac{1}{R_1C_1R_2C_2s^2 + (R_1C_1 + R_2C_2 + R_2C_1)s + 1}\right)$$

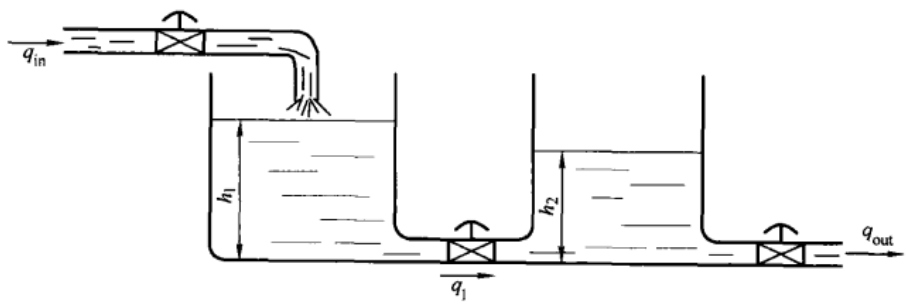


图 2-52　题 2-12 双水箱系统原理图

第三章　控制系统数字仿真的实现

第一节　控制系统的结构及其拓扑描述

一、控制系统常见的典型结构形式

控制系统结构形式往往很复杂，但归结起来，不外乎以下几种类型。

（一）单输入-单输出开环控制结构（见图 3-1）

控制器、控制对象等环节可用第二章所述任何一种数学模型描述。图 3-1 中，$r(t)$ 为参考输入量；$u(t)$ 为控制量；$y(t)$ 为输出量。

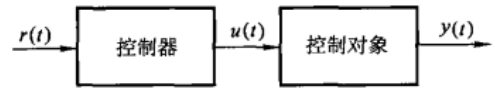


图 3-1　单输入-单输出开环控制结构图

（二）单输入-单输出前馈控制结构（见图 3-2）

这种结构形式本身仍为开环控制形式，但为补偿输入引起的误差，在已知误差变化规律情况下，加入补偿环节，对误差作提前修正。

补偿环节同样可用任一种数学模型描述，其余同图 3-1。

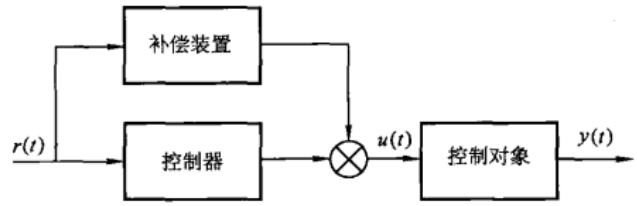


图 3-2　单输入-单输出前馈控制结构图

（三）单输入-单输出闭环（反馈）控制结构（见图 3-3）

这是控制系统中应用最广泛的控制结构形式，大多数控制系统为保证相应的控制精度、达到要求的性能指标，都采用这种闭环负反馈形式（如图中所示）。

其原理主要是根据偏差（或误差）确定控制量，使输出量按期望的精度变化，减小或消除偏差，许多参考文献又称其为偏差控制系统。图 3-3 中，$e(t)$ 为偏差量（或误差量）；$b(t)$ 为反馈量。

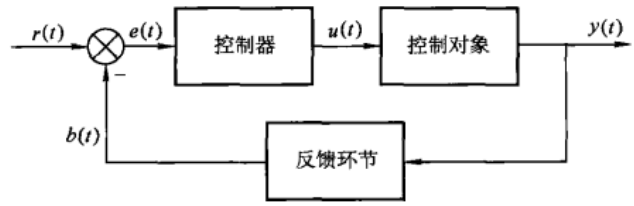


图 3-3　单输入-单输出闭环控制结构图

反馈环节可用任一种数学模型描述。当反馈环节为输出量直接反馈形式时

$$b(t) = y(t)$$

则偏差

$$e(t) = r(t) - b(t) = r(t) - y(t)$$

称做误差，而此时的闭环系统称为单位反馈系统。

凡是单输入-单输出控制系统结构均能方便地表示为图论中的拓扑结构形式，即使系统环节再增多，反馈和前馈联系再复杂，表述也很清晰，如图 3-4 所示，图中各环节用序号表

示，每环节都有自己的输入和输出变量 u_i 和 y_i（$i=1, 2, \cdots, 6$），对应其相互关系的数学描述，可以是第二章所述形式的任一种。

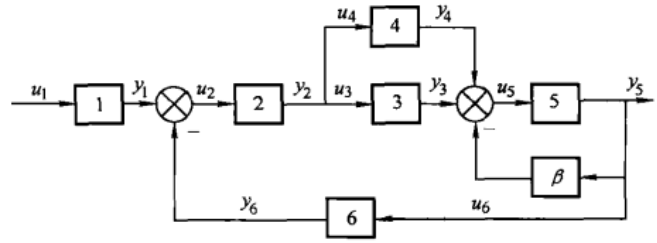


图 3-4　控制系统拓扑结构图

（四）多输入-多输出控制结构

若干个单输入-单输出控制结构形式通过一定方式组合在一起，就构成了多输入-多输出控制结构形式，用来描述较复杂的多变量控制系统。

根据不同的环节组合形式，也有不同的控制结构。如变量和环节有单向耦合作用关系的结构形式（见图 3-5a）、交叉反馈耦合作用关系的结构形式（见图 3-5b）等。

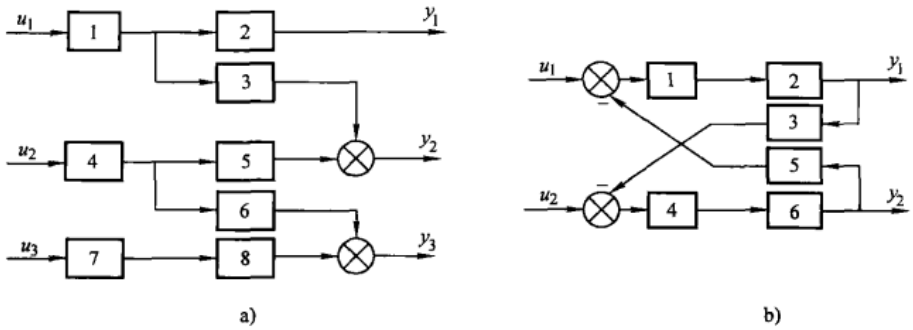


图 3-5　多输入-多输出控制结构图

a）单向耦合结构　b）交叉反馈耦合结构

同样，由图可知，用图论拓扑结构描述这类复杂系统非常简洁清楚。我们在后面的分析中将会更深入地体会到，这样描述给系统仿真带来极大的方便。

二、控制系统的典型环节描述

任何一个复杂的线性控制系统，都是由一些简单的不同类型的具体线性环节组合而成的，若对常见的一些简单线性环节能准确地加以定量描述，则复杂线性系统的描述也只是复杂在各部分的相互连接关系上。采用上节所述拓扑关系描述，并把这种连接关系用相应的数学关系表达出来，就可以得到在计算机上能方便运行的“二次模型”。首先来看在经典控制理论中常见的典型环节。

（一）比例环节

$$G_i(s)=\frac{y_i}{u_i}=K_i$$

（二）惯性环节

$$G_i(s)=\frac{y_i}{u_i}=\frac{K_i}{T_i s+1}$$

（三）惯性比例环节

$$G_i(s)=\frac{y_i}{u_i}=\frac{K_i(\tau_i s+1)}{T_i s+1}$$

（四）积分环节

$$G_i(s)=\frac{y_i}{u_i}=\frac{K_i}{T_i s}$$

（五）积分比例环节

$$G_i(s)=\frac{y_i}{u_i}=\frac{K_i(\tau_i s+1)}{T_i s}$$

（六）二阶振荡环节

$$G_i(s)=\frac{y_i}{u_i}=\frac{K_i\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2}=\frac{K_i}{T_i^2 s^2+2\xi T_i s+1}$$

式中，ξ 为阻尼比，在二阶振荡情况下，$0\leqslant\xi<1$；ω_n 为无阻尼自然振荡频率；T_i 为无阻尼自然振荡周期，$T_i=1/\omega_n$。

（七）高阶线性环节

$$G_i(s)=\frac{y_i}{u_i}=\frac{b_0 s^m+\cdots+b_{m-1}s+b_m}{a_0 s^n+\cdots+a_{n-1}s+a_n}$$

式中，a_0，$a_1\cdots$，a_n 为分母多项式系数，亦称特征多项式系数；b_0，b_1，…，b_m 为分子多项式系数；它们均为反映高阶线性环节动态响应性能的重要参数。

对以上单个典型线性环节，无论其阶次高低，均可应用 MATLAB 语言方便地求出在输入量 u_i 作用下，输出量 y_i 的响应性能。y_i 和 u_i 之间的数学表达关系可以由传递函数 G_i（s）方便地转化为其他各种形式，如状态方程、零极点增益、部分分式等。

三、控制系统的连接矩阵

对图 3-4 所表示的线性系统，各环节均为线性的，在各自的输入量 u_i 作用下，给出各自的输出量 y_i，这种作用关系是通过各环节的数学描述体现出来的，其数学关系可以是上节所述任何一种。但各环节之间存在相互作用，u_i 不是孤立的，只要与其他环节有连接关系，就要受到相应 y_i 变化的影响。因此，要完整地将系统描述出来，还应该分析各环节输出 y_i 对其他环节有无输入作用，才能完整地进行仿真分析。

根据图 3-4 中 u_i、y_i 拓扑连接关系，可逐个写出每个环节输入 u_i 受哪些环节输出 y_i 的制约和影响，现列写如下

$$\begin{cases}u_1=r\\u_2=y_1-y_6\\u_3=y_2\\u_4=y_2\\u_5=y_3+y_4-\beta y_5\\u_6=y_5\end{cases}\tag{3-1}$$

由式中可见，除 u_1 只与参考输入 r 有直接联系外，其余各环节输入 u_i 都有可能与其他环节输出 y_i 有关。表示为如下形式，可以更为清楚地观察各环节的连接关系

$$\begin{cases} u_1 = 0 \cdot y_1 + 0 \cdot y_2 + 0 \cdot y_3 + 0 \cdot y_4 + 0 \cdot y_5 + 0 \cdot y_6 + 1 \cdot r \\ u_2 = 1 \cdot y_1 + 0 \cdot y_2 + 0 \cdot y_3 + 0 \cdot y_4 + 0 \cdot y_5 - 1 \cdot y_6 + 0 \cdot r \\ u_3 = 0 \cdot y_1 + 1 \cdot y_2 + 0 \cdot y_3 + 0 \cdot y_4 + 0 \cdot y_5 + 0 \cdot y_6 + 0 \cdot r \\ u_4 = 0 \cdot y_1 + 1 \cdot y_2 + 0 \cdot y_3 + 0 \cdot y_4 + 0 \cdot y_5 + 0 \cdot y_6 + 0 \cdot r \\ u_5 = 0 \cdot y_1 + 0 \cdot y_2 + 1 \cdot y_3 + 1 \cdot y_4 - \beta \cdot y_5 + 0 \cdot y_6 + 0 \cdot r \\ u_6 = 0 \cdot y_1 + 0 \cdot y_2 + 0 \cdot y_3 + 0 \cdot y_4 + 1 \cdot y_5 + 0 \cdot y_6 + 0 \cdot r \end{cases}$$

由式可见，按信号传递的方向，凡与其他环节没有连接关系的环节，其输出 y_i 的系数均为0；凡与其他环节有连接关系的环节，其输出 y_i 的系数不全为0；凡与参考输入 r 连接的环节，r 的系数不为0。而 r 的系数为0，则表示参考输入 r 不与该环节相连。

把环节之间的关系和环节与参考输入的关系分别用矩阵表示出来，成为

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & -1 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & -\beta & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} r$$

即

$$\boldsymbol{U} = \boldsymbol{W}\boldsymbol{Y} + \boldsymbol{W}_0 r$$

式中，$\boldsymbol{W}$ 为连接矩阵，$n \times n$ 型，阵中元素清楚地表示出各环节之间的拓扑连接关系；$\boldsymbol{W}_0$ 为输入连接矩阵，$n \times 1$ 型（当参考输入为1维时），阵中元素表示环节与参考输入之间的拓扑连接关系；$\boldsymbol{U} = [u_1, u_2, \cdots, u_n]^{\mathrm{T}}$，为各环节输入向量；$\boldsymbol{Y} = [y_1, y_2, \cdots, y_n]^{\mathrm{T}}$，为各环节输出向量。

仔细研究连接矩阵 $\boldsymbol{W}$，可从其元素值直接看出各环节之间的连接情况，即

$w_{ij} = 0$，环节 j 不与环节 i 相连；

$w_{ij} \neq 0$，环节 j 与环节 i 有连接关系；

$w_{ij} > 0$，环节 j 与环节 i 直接相连（$w_{ij} = 1$）或通过比例系数相连（w_{ij}为任意正实数）；

$w_{ij} < 0$，环节 j 与环节 i 直接负反馈相连（$w_{ij} = -1$）或通过比例系数负反馈相连（w_{ij}为任意负实数）；

$w_{ii} \neq 0$，环节 i 单位自反馈（$w_{ii} = 1$ 或 $w_{ii} = -1$）或通过比例系数自反馈（w_{ii}为任意实数）。

以连接矩阵表示复杂系统中各环节的连接关系，使得对复杂连接结构的控制系统的仿真变得简明方便。同样的思路完全可以用在连接关系更为复杂的多输入、多输出控制系统中，只需将与参考输入、系统输出有关系的矩阵维数扩展到与输入、输出向量维数一致即可。

连接矩阵建立起来后，对表述为结构图形式的控制系统进行仿真尤为方便。结构图中每一环节的数学描述可以是前节所述典型环节的任一种描述，甚至可以拓宽表述范围到非线性环节、离散（采样）环节、数学控制环节。MATLAB 语言环境下的工具软件 SIMULINK 是这种仿真思路的典型体现。软件中以模块库形式提供了非常齐全的环节形式，包含了控制系统常用的各类环节，如 Continuous 库中从低阶到高阶的各种线性环节，Discontinuities 库中的

常见非线性环节（滞后、饱和、死区等）；Discrete 库中的各类离散化、数字化环节等。

第二节 面向系统结构图的数字仿真

实际工程中常常给出的是结构图形式的控制系统数学模型，对此类形式的系统进行仿真分析，主要是根据“二次模型”编写适当程序语句，使之能自动求解各环节变量的动态变化情况，从而得到关于系统输出各变量的有关数据、曲线等，以对系统进行性能分析和设计。

一、典型闭环系统的数字仿真

本节先对控制系统最常见的典型结构形式二次模型化，然后讨论采用数值积分算法求解系统响应的仿真程序实现。

（一）典型闭环系统结构形式

控制系统最常见的典型闭环系统结构，如图 3-6 所示，且有

$$G(s)=\frac{Y(s)}{U(s)}=\frac{b_0s^m+\cdots+b_{m-1}s+b_m}{a_0s^n+\cdots+a_{n-1}s+a_n},\quad U(s)=E(s)$$

它是在微分方程式（2-1）基础上，经零初始条件下的拉普拉斯变换求得的，也就是式（2-4）。在这里 $G(s)$ 仅表示系统的开环传递函数，描述控制量 $u(t)$ 与输出量 $y(t)$ 间的信号传递关系。

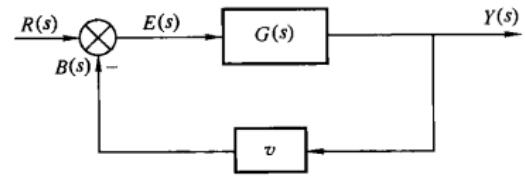


图 3-6 典型闭环系统结构图

v 是系统的反馈系数，设其为一常系数。它的大小反映了反馈量 $b(t)$ 与输出量 $y(t)$ 之间的比例关系。

（二）系统仿真模型与求解思路

所谓仿真模型是指经一定方式把数学模型转化为便于在计算机上运行的表达形式。这种表达形式往往是一些适合于具体编程实现的数学关系描述式。

对图 3-6 系统的开环传递函数 $G(s)$，可按照能控标准型写出其开环状态方程

$$\begin{cases}\dot{\boldsymbol{X}}=\boldsymbol{AX}+\boldsymbol{BU}\\ \boldsymbol{Y}=\boldsymbol{CX}\end{cases}\tag{3-2}$$

式中

$$\boldsymbol{A}=\begin{bmatrix}0&1&0&\cdots&\cdots&0\\0&0&1&\cdots&\cdots&0\\\vdots&\vdots&\vdots&&&\\&&&&&1\\-\bar{a}_n&&\cdots&\cdots&&-\bar{a}_1\end{bmatrix};\quad \boldsymbol{B}=\begin{bmatrix}0\\0\\\cdots\\0\\1\end{bmatrix};$$

$$\boldsymbol{C}=[\,\bar{b}_m\quad\bar{b}_{m-1}\quad\cdots\quad\bar{b}_0\quad0\quad\cdots\quad0\,];$$

注意：$\boldsymbol{A}$，$\boldsymbol{C}$ 阵中 $\bar{a}_j$，$\bar{b}_i$（$j=1,2,\cdots,n$；$i=0,1,\cdots,m$）为对式（2-4）分母首一化后分母、分子各系数，即 $\bar{a}_j=\dfrac{a_j}{a_0}$，$\bar{b}_i=\dfrac{b_i}{b_0}$，且 $m=n-1$。

由图又知，控制量 $\boldsymbol{U}=r-v\boldsymbol{Y}$，代入式（3-2）得

$$\dot{\boldsymbol{X}}=\boldsymbol{A}\boldsymbol{X}+\boldsymbol{B}(r-v\boldsymbol{Y})$$

再由

$$\boldsymbol{Y}=\boldsymbol{C}\boldsymbol{X}$$

则

$$\dot{\boldsymbol{X}}=(\boldsymbol{A}-\boldsymbol{B}v\boldsymbol{C})\boldsymbol{X}+\boldsymbol{B}r=\boldsymbol{A}_b\boldsymbol{X}+\boldsymbol{B}r \tag{3-3}$$

即得系统闭环状态方程，其中

$$\boldsymbol{A}_b=\boldsymbol{A}-\boldsymbol{B}v\boldsymbol{C}$$

为系统闭环系数矩阵，而输入矩阵 $\boldsymbol{B}$ 和输出矩阵 $\boldsymbol{C}$ 不变。这就是图 3-6 系统的仿真模型。

仿真模型一旦确立，就可以着手考虑求解与编程实现。观察式（3-3）可知，该式其实为一个一阶微分方程组的矩阵表达形式，而数值积分法最适宜解一阶微分方程，当采用四阶龙格-库塔法求解此闭环状态方程时，其步骤如下：

对

$$\dot{\boldsymbol{X}}=\boldsymbol{A}_b\boldsymbol{X}+\boldsymbol{B}r$$

知

$$f(t,\ \boldsymbol{X})=\boldsymbol{A}_b\boldsymbol{X}+\boldsymbol{B}r \tag{3-4}$$

此为对应 n 个状态变量 $\boldsymbol{X}=[x_1,\ x_2,\ \cdots,\ x_n]^{\mathrm{T}}$ 一阶导数 $\dot{\boldsymbol{X}}$ 的 n 维向量表达式。其中，r 为随时间变化的已知输入函数。于是，当求解过程进行到 $t=t_k$ 时刻，欲求 t_{k+1} 时刻各量，需先求

第一斜率：　$\boldsymbol{K}_1=f(t_k,\ \boldsymbol{X}_k)=\boldsymbol{A}_b\boldsymbol{X}_k+\boldsymbol{B}r(t_k)$

再求第二斜率：　$\boldsymbol{K}_2=f\left(t_k+\frac{h}{2},\ \boldsymbol{X}_k+\frac{h}{2}\boldsymbol{K}_1\right)=\boldsymbol{A}_b\left(\boldsymbol{X}_k+\frac{h}{2}\boldsymbol{K}_1\right)+\boldsymbol{B}r\left(t_k+\frac{h}{2}\right)$

同样得第三斜率：　$\boldsymbol{K}_3=f\left(t_k+\frac{h}{2},\ \boldsymbol{X}_k+\frac{h}{2}\boldsymbol{K}_2\right)=\boldsymbol{A}_b\left(\boldsymbol{X}_k+\frac{h}{2}\boldsymbol{K}_2\right)+\boldsymbol{B}r\left(t_k+\frac{h}{2}\right)$

第四斜率：　$\boldsymbol{K}_4=f(t_k+h,\ \boldsymbol{X}_k+h\boldsymbol{K}_3)=\boldsymbol{A}_b(\boldsymbol{X}_k+h\boldsymbol{K}_3)+\boldsymbol{B}r(t_k+h)$

以上各斜率 $\boldsymbol{K}_1$、$\boldsymbol{K}_2$、$\boldsymbol{K}_3$、$\boldsymbol{K}_4$ 可认为是对应 n 维状态变量 $\boldsymbol{X}=[x_1,\ x_2,\ \cdots,\ x_n]^{\mathrm{T}}$ 在 $t=t_k$ 时刻的四组斜率，每组为 n 维斜率向量，即每组 n 个龙格-库塔系数。

最后，再由

$$\boldsymbol{X}_{k+1}=\boldsymbol{X}_k+\frac{h}{6}(\boldsymbol{K}_1+2\boldsymbol{K}_2+2\boldsymbol{K}_3+\boldsymbol{K}_4)$$

求得 t_{k+1} 时刻状态 $\boldsymbol{X}_{k+1}$，即可得输出相应时刻值

$$y_{k+1}=\boldsymbol{C}\boldsymbol{X}_{k+1}$$

按以上算式，取 $k=0,\ 1,\ 2,\ \cdots,\ N$ 不断递推，即求得所需时间 $t_0,\ t_1,\ \cdots,\ t_N$ 各点的状态变量 $\boldsymbol{X}(t_k)$ 和输出量 $\boldsymbol{Y}(t_k)$。

（三）仿真程序框图与实现

基于以上仿真模型和求解思路，可以着手考虑面向这类典型闭环控制系统结构图的仿真程序的编制与实现。

作为系统仿真程序，使用时应尽可能方便，使用者只要将开环传递函数 $G(s)$ 的分母、分子各系数 a_j（$j=0,\ 1,\ 2,\ \cdots,\ n$），b_i（$i=0,\ 1,\ 2,\ \cdots,\ n-1$）和反馈系数 v 输入计算机，计算机就掌握了关于该系统的基本信息——模型参数。然后求取首一化表达形式，以及形成开、闭环状态方程各阵等步骤均由仿真程序自动完成，无需人工干预。因此，程序应有输入数据模块和初始化程序模块。

在参考输入函数 $r(t)$ 的作用下，系统输出 $y(t)$ 开始随时间变化，仿真程序应能按照给定的计算步长，采用已确定的数值算法，对系统中各状态变量和输出逐点变化情况进行求解运算。这部分模块是整个仿真程序的核心部分，计算速度、精度误差均取决于它。通常称为运行程序模块。

在使用者规定的时间范围内，将计算数据按照一定要求存储，并在仿真结束时，按使用者指定格式输出仿真结果，以便对系统进行分析研究，这就是所谓的输出程序模块。

综上所述，构成一个完整的仿真程序，必须至少建立：输入数据块、初始化块、运行计算块、输出结果块四个子程序模块，才能正常地完成仿真任务。

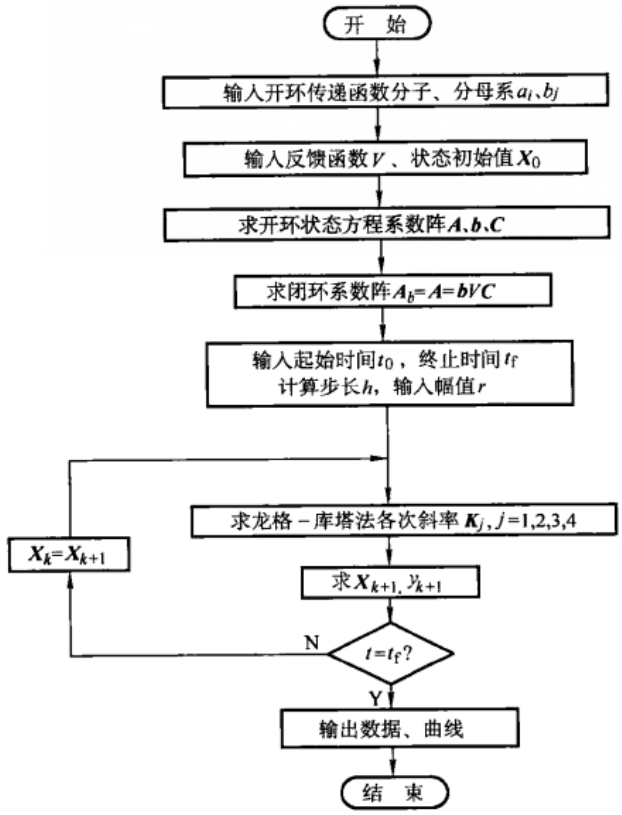


图 3-7 典型闭环系统的仿真程序框图

(1) 程序框图 程序框图如图 3-7 所示。按照框图，采用任何形式的高级语言都可编程实现。本书中均采用 MATLAB 语言编程实现。该语言最大的优势是具有非常强大的矩阵处理功能，编写系统仿真程序时，可以不必考虑一些基本的矩阵运算如何实现，大大节省了编程者的精力和时间，从而可以更多地考虑如何针对各类控制系统的特点得到相应的仿真实现。

(2) 程序语句 用 MATLAB 语言编制的程序语句如下：

1) 输入数据：

```
%---------- filename:sp3_1.m 面向典型闭环系统的仿真程序----------
a = [a0,a1,……,an];                    % n+1 维分母系数向量
b = [b0,b1,……,bm];                    % m+1 维子系数向量
X0 = [x10,x20,……,xn0];                 % 状态向量初值
V = v0;                                % 反馈系数
n = n0;                                % 系统阶次
T0 = t0;                               % 起始时间
Tf = tf;                               % 终止时间
h = h0;                                % 计算步长
R = r;                                 % 阶跃输入函数的幅值
```

2）形成开、闭环系数矩阵：

```
b = b/a(1);a = a/a(1);A = a(2: n+1);                    % 首一化处理
A = [rot90(rot90(eye(n-1,n)));-fliplr(A)];              % 形成能控标准型 A 阵
B = [zeros(1,n-1),1]';                                  % 形成输入阵 B(n 维列向量)
m1 = length(b);                                         % 分子系数向量维数 m+1
C = [fliplr(b),zeros(1,n-m1)];                          % 形成输出阵 C(n 维行向量)
Ab = A-B*C*V;                                           % 形成闭环系数阵 Ab
X = X0';y = 0;t = T0;                                   % 设初值,准备开始递推运算
```

3）运算求解：

```
N = round(Tf-T0)/h;                                     % 确定输出点数
for i = 1: N                                            % 四阶龙格—库塔法
  K1 = Ab*X+B*R
  K2 = Ab*(X+h*K1/2)+B*R;
  K3 = Ab*(X+h*K2/2)+B*R;
  K4 = Ab*(X+h*K3)+B*R;                                 % 求各次斜率 K
  X = X+h*(K1+2*K2+2*K3+K4)/6;                          % 求状态
  y = [y,C*X];                                          % 求输出并以向量形式保存
  t = [t,t(i)+h];                                       % 输出对应时刻以向量形式保存
end
```

4）输出结果：

```
[t',y']                                                 % 输出数据形式结果
plot(t,y)                                               % 输出曲线形式结果
```

（注：程序中变量名不能带上、下标；为清楚起见，语句中字母、数字均为正体，斜体部分表示被输入的具体数据）

以上即为采用 MATLAB 语言实现面向典型闭环系统数字仿真的程序，从第 2）部分到程序末尾可编辑为 sp3_1.m 文件存储起来。使用时，只要进入到 MATLAB 语言环境，按第 1）部分格式输入系统参数和运行参数，再调用该文件，即可得运算仿真结果。

上面程序中用到了几个 MATLAB 语言的特殊功能，使得程序较为简练。简要说明如下：

eye(m,n)；　　产生 $m \times n$ 型 I 阵，即主对角元素均为 1，其余元素为零的长方阵

length(　)；　　求取括号内向量的维数

zeros(m,n)；　　产生 $m \times n$ 型零阵

rot90(　)；　　将括号内矩阵左旋 90°

fliplr(　)；　　将括号内矩阵或向量左右翻转

round(　)；　　对括号内运算取整

plot(t,y)；　　打印以 t 为横轴，y 为纵轴的 $y(t)$ 曲线

在后续内容中，将有更多的特殊功能被用到，本书将结合实际程序逐步予以介绍。

本程序为清晰描述如何编程实现面向典型闭环结构的系统仿真，没有运用更多技巧，而完全按照前述仿真实现思路，原理性地安排语句，以便于读者理解和消化内容。后面各章节中程序编制均以此为基本思路，不再解释说明。随着对 MATLAB 语言的了解逐步深入，相

信读者会编制出输入、输出界面漂亮，人机交互方便、通用性更强的程序。

下面举一实例，应用程序 sp3_1.m 求解系统输出响应。

例 3-1 求图 3-8 所示系统的阶跃响应 $y(t)$ 的数值解。

解 该系统结构形式为典型闭环控制系统，用 sp3_1.m 求解过程如下：

1）取开环放大系数 $k=1$，反馈系数 $v=1$（单位反馈系统），阶跃输入幅值 $r=1$。

2）利用 MATLAB 语言中 conv（ ）卷积函数功能，先将系统开环传递函数 $G(s)$ 化为式（2-4）传递函数形式的分母、分子多项式系数向量 $[a_0, a_1, \cdots, a_n]$ 和 $[b_0, b_1, \cdots, b_m]$。

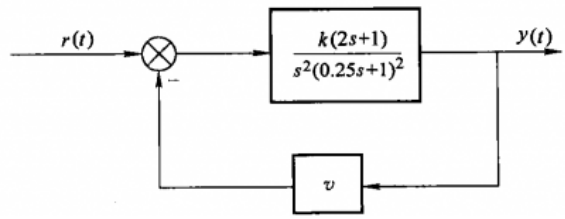


图 3-8 例 3-1 控制系统结构图

3）设系统状态向量初值 $[x_{10}, x_{20}, \cdots, x_{n0}]$ 均为零。

4）系统运行参数 $n_0=4$，$t_0=0$，$t_f=10$，$h_0=0.25$。

5）按以上步骤和参数，在 MATLAB 语言环境下，输入以下命令语句（“>>”表示在 MATLAB 命令窗口环境）。

```
>> k = 1;
>> a = conv([1  0  0],conv([0.25  1],[0.25  1]))
a =
    0.0625   0.5000   1.0000        0        0
>> b = [2 * k  k]
b =
     2     1
>> X0 = [0  0  0  0]
X0 =
     0     0     0     0
>> V = 1;n = 4;T0 = 0;Tf = 10;h = 0.25;R = 1;
>> sp3_1                    % 运行 sp3_1.m 仿真程序
>> [t',y']                  % 输出数据形式结果(见表 3-1)
```

表 3-1 例 3-1 输出响应 $y(t)$ 数据

t	$y(t)$	t	$y(t)$	t	$y(t)$	t	$y(t)$
0.25	0.0443	2.75	1.0619	5.25	1.0776	7.75	1.0043
0.50	0.2779	3.00	0.9399	5.50	1.0589	8.00	1.0095
0.75	0.6392	3.25	0.8715	5.75	1.0345	8.25	1.0118
1.00	1.0237	3.50	0.8563	6.00	1.0106	8.50	1.0113
1.25	1.3410	3.75	0.8826	6.25	0.9919	8.75	1.0088
1.50	1.5361	4.00	0.9333	6.50	0.9809	9.00	1.0053
1.75	1.5944	4.25	0.9905	6.75	0.9778	9.25	1.0018
2.00	1.5342	4.50	1.0397	7.00	0.9811	9.50	0.9990
2.25	1.3950	4.75	1.0718	7.25	0.9883	9.75	0.9972
2.50	1.2235	5.00	1.0837	7.50	0.9968	10.00	0.9966

同时输出曲线形式结果（见图 3-9）。

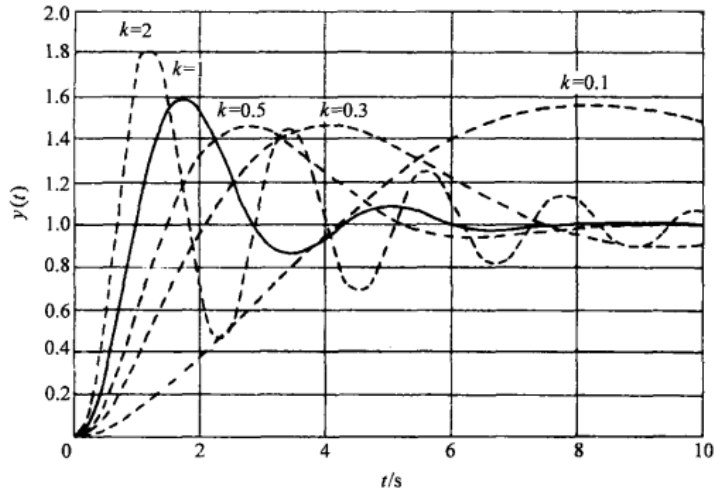


图 3-9　例 3-1 输出响应 $y(t)$ 曲线

若要分析开环放大系数 k、反馈系数 v 的影响作用，只需分别对 k、v 重新赋值，其余参数不变，再次运行 sp3_1. m 即可。图 3-9 中示出了 k 取不同值时输出响应 $y(t)$ 的曲线。由所示曲线清楚地看到 k 增大使系统振荡加剧，甚至可能造成不稳定；而 k 减小将对系统快速性和稳态精度有很大影响。

二、复杂连接的闭环系统数字仿真

实际工程中常常遇到的是复杂结构形式的控制系统，它由若干典型环节按照一定规律连接而成，可用前述的拓扑形式将其描述为复杂连接的闭环系统结构图。要采用上节所述方法仿真，必须先将复杂形式结构图简化成为图 3-6 所示的典型结构形式，求出开环传递函数，然后运用上节程序进行仿真分析。但存在如下问题：①系统结构复杂，内部存在交叉耦合、局部回环等情况时，化简并非易事，尤其通过手工化简更使工作量陡增；②有时在分析中，还需要得知结构图中某些环节的输出变量情况，若经化简消去这些环节，则不便进行观察分析；③在分析中常常需要改变某参数，观察其对输出的影响，但改动一个参数值，就有可能需将所有开环传递函数分子、分母系数统统重新计算再次输入计算机，很不方便；④对实际系统中存在的非线性情况，更是无法加以考虑。鉴于以上原因，编制实现面向复杂连接的闭环系统结构图的仿真程序时，必须使其能克服上节仿真程序的不足之处，应该具有以下特点：

1）可按照系统结构图输入各环节参数，对应关系明确，改变参数方便。

2）可方便地观察各环节输出动态响应。

3）各环节存在非线性特性时易于处理。

（一）典型环节的二次模型化

复杂连接闭环系统数字仿真的基本思路是：与实际系统的结构图相对应，在计算机程序中也应构出方便表示各实际环节的典型环节，并将环节之间的连接关系输入计算机，由计算机程序自动形成闭环状态方程，运用数值积分方法求解响应。

由此可知，选定典型环节是很重要的，要使其既具有代表性，又不致于造成输入数据复

杂繁琐。考察控制系统常见环节，可有如下几种情况：

比例环节：$G(s)=K$

积分环节：$G(s)=\dfrac{K}{s}$

比例积分：$G(s)=K_1+\dfrac{K_2}{s}=\dfrac{K_1 s+K_2}{s}$

惯性环节：$G(s)=\dfrac{K}{Ts+1}$

一阶超前滞后环节：$G(s)=K\dfrac{T_1 s+1}{T_2 s+1}$

二阶振荡环节：$G(s)=\dfrac{K}{T^2 s^2+2\xi Ts+1}$

可见，除二阶振荡环节外，都是一阶环节，完全可用一个通用一阶环节（见图 3-10）表示，即

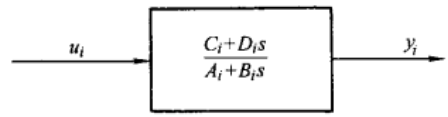


图 3-10 典型一阶环节

$$G_i(s)=\frac{y_i(s)}{u_i(s)}=\frac{C_i+D_i s}{A_i+B_i s}\quad(i=1,2,\cdots,n)\tag{3-5}$$

式中，$y_i(s)$ 为环节 i 的输出；$u_i(s)$ 为环节 i 的输入；n 为系统中的环节数（也就是系统的阶次）。而二阶振荡环节，可以化为用图 3-11 所示方式连接而成的等效结构图。

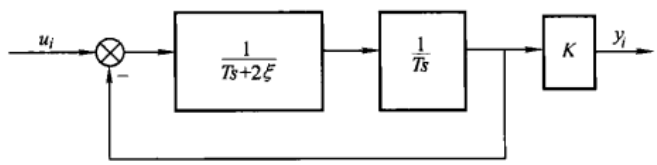


图 3-11 二阶振荡环节的等效结构图

由图可见，二阶振荡环节完全可以用一阶环节等效连接得到。因此，选定式（3-5）作为组成复杂系统仿真模型的典型环节是适合的。

设：输入向量 $\boldsymbol{U}=[u_1,u_2,\cdots,u_n]^{\mathrm{T}}$，其中各分量表示各环节输入量；

输出向量 $\boldsymbol{Y}=[y_1,y_2,\cdots,y_n]^{\mathrm{T}}$，各分量表示各环节输出量；

模型参数阵为

$$\boldsymbol{A}=\begin{bmatrix}A_1 & & 0\\ & A_2 & \\ & & \ddots \\ 0 & & A_n\end{bmatrix};\quad \boldsymbol{B}=\begin{bmatrix}B_1 & & 0\\ & B_2 & \\ & & \ddots \\ 0 & & B_n\end{bmatrix};$$

$$\boldsymbol{C}=\begin{bmatrix}C_1 & & 0\\ & C_2 & \\ & & \ddots \\ 0 & & C_n\end{bmatrix};\quad \boldsymbol{D}=\begin{bmatrix}D_1 & & 0\\ & D_2 & \\ & & \ddots \\ 0 & & D_n\end{bmatrix}。$$

于是系统中所有环节输出、输入关系统一用矩阵表示如下

$$(\boldsymbol{A}+\boldsymbol{B}s)\ \boldsymbol{Y}=(\boldsymbol{C}+\boldsymbol{D}s)\ \boldsymbol{U}\tag{3-6}$$

（二）系统的连接矩阵与仿真求解

1. 连接矩阵

仅把系统中各环节描述出来还不够，要进行数值积分求解，还必须把各环节之间的相互作用关系清楚地表达出来，这种表达是通过建立本章第一节所述的连接矩阵实现的。

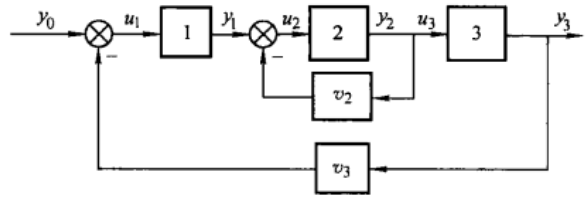


图 3-12　某闭环系统拓扑结构图

为说明问题，现举例说明如下：

设系统结构如图 3-12 所示，图中编号 1、2、3 表示三个典型环节，v_2、v_3 为环节间直接作用的比例系数。各环节输入 u_i 与输出 y_i 关系为

$$\begin{cases} u_1 = y_0 - v_3 y_3 \\ u_2 = y_1 - v_2 y_2 \\ u_3 = y_2 \end{cases}$$

整理为矩阵形式得

$$\boldsymbol{U} = \boldsymbol{W}\boldsymbol{Y} + \boldsymbol{W}_0 y_0 \tag{3-7}$$

式中

$$\boldsymbol{W} = \begin{bmatrix} 0 & 0 & -v_3 \\ 1 & -v_2 & 0 \\ 0 & 1 & 0 \end{bmatrix}; \quad \boldsymbol{W}_0 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \tag{3-8}$$

典型环节和连接矩阵确定后，通过进一步推导闭环状态方程，才能得到完整的仿真模型，以便利用前述的数值积分递推公式计算求解。

2. 系统的求解

将式（3-7）代入式（3-6），则

$$(\boldsymbol{A} + \boldsymbol{B}s)\ \boldsymbol{Y} = (\boldsymbol{C} + \boldsymbol{D}s)(\boldsymbol{W}\boldsymbol{Y} + \boldsymbol{W}_0 y_0)$$

整理，得

$$(\boldsymbol{B} - \boldsymbol{D}\boldsymbol{W})\ s\boldsymbol{Y} = (\boldsymbol{C}\boldsymbol{W} - \boldsymbol{A})\ \boldsymbol{Y} + \boldsymbol{C}\boldsymbol{W}_0 y_0 + \boldsymbol{D}\boldsymbol{W}_0 s y_0$$

简洁表达为

$$\boldsymbol{Q}s\boldsymbol{Y} = \boldsymbol{R}\boldsymbol{Y} + \boldsymbol{V}_1 y_0 + \boldsymbol{V}_2 s y_0 \tag{3-9}$$

式中

$$\boldsymbol{Q} = \boldsymbol{B} - \boldsymbol{D}\boldsymbol{W}$$
$$\boldsymbol{R} = \boldsymbol{C}\boldsymbol{W} - \boldsymbol{A}$$
$$\boldsymbol{V}_1 = \boldsymbol{C}\boldsymbol{W}_0$$
$$\boldsymbol{V}_2 = \boldsymbol{D}\boldsymbol{W}_0$$

若 $\boldsymbol{Q}$ 阵的逆存在，则式（3-9）两边同时左乘 $\boldsymbol{Q}^{-1}$，得

$$s\boldsymbol{Y} = \boldsymbol{Q}^{-1}\boldsymbol{R}\boldsymbol{Y} + \boldsymbol{Q}^{-1}\boldsymbol{V}_1 y_0 + \boldsymbol{Q}^{-1}\boldsymbol{V}_2 s y_0$$

两边进行拉普拉斯反变换，求得系统闭环状态方程时域表达式为

$$\dot{\boldsymbol{Y}} = \boldsymbol{A}_b \boldsymbol{Y} + \boldsymbol{b}_1 y_0 + \boldsymbol{b}_2 \dot{y}_0 \tag{3-10}$$

其中，闭环状态各变量即为各环节输出变量。而 $\boldsymbol{A}_b = \boldsymbol{Q}^{-1}\boldsymbol{R}$；$\boldsymbol{b}_1 = \boldsymbol{Q}^{-1}\boldsymbol{V}_1$；$\boldsymbol{b}_2 = \boldsymbol{Q}^{-1}\boldsymbol{V}_2$ 为闭环系统的系数阵和输入阵。

式（3-10）是典型的一阶微分方程组矩阵形式，利用前节介绍的求解方法，可方便地求出各环节的输出响应。但应注意该式右端中多出一项参考输入的导数项，为不增加求解的复杂性，应设法去掉该项。因此，建立该系统仿真模型的过程中应注意以下两点：

（1）保证 $\boldsymbol{Q}$ 阵有逆　$\boldsymbol{Q}$ 若不存在逆阵，则整个推导过程无意义，系统无法求解。通过分析 $\boldsymbol{Q}$ 阵的构成可知：当环节为纯比例或纯微分环节时，有可能使 $\boldsymbol{Q}$ 阵中某行全为零，造成 $\boldsymbol{Q}$ 奇异而无法求逆。因此，只要严格按 $B_i \neq 0$ 原则确定一个环节，则系统闭环过程中 $\boldsymbol{Q}$ 阵必有逆，计算得以正常进行。若系统中出现纯比例、纯微分环节，则应设法与其他环节合并处理，或设法化为系统可接受的环节。

（2）去掉 $\dot{y}_0$ 项　当参考输入量为 $y_0 = 1$（t）时，则 $\dot{y}_0 = 0$，那么按以上思路计算是准确的。但为任意参考输入量时，要保证一定的计算精度，则式（3-10）中 $\dot{y}_0$ 项不可避免地存在，而在实际计算中 $\dot{y}_0$ 又不易准确表达，近似表达又影响精度，故欲使 $\dot{y}_0$ 项不出现，只有使相应系数阵 $\boldsymbol{V}_2 = 0$。由式（3-9）知

$$\boldsymbol{V}_2 = \boldsymbol{D}\boldsymbol{W}_0$$

其中，$\boldsymbol{W}_0$ 为外加输入对各环节的作用关系阵，在单变量情况下，实际为列向量形式，不可能为零，因此要使 $\boldsymbol{V}_2 = 0$，只有来考虑 $\boldsymbol{D}$ 中元素 $\boldsymbol{D}_i$（$i = 1, 2, \cdots, n$）。当 $\boldsymbol{W}_0$ 中元素 $w_{i0} = 0$，则相应环节 i 的 D_i 可以不为零；当 $\boldsymbol{W}_0$ 中元素 $w_{i0} \neq 0$，则相应环节 i 的 $\boldsymbol{D}_i$ 必须为零。这样就可使 $\boldsymbol{V}_2 = \boldsymbol{D}\boldsymbol{W}_0 = 0$。

综上可知，只要在建立系统模型参数时，注意使含有微分项系数的环节不直接与外加参考输入连接，即可避免在递推计算公式中出现 $\dot{y}_0$ 项，达到求解方便的目的。

3. 仿真程序框图与实现

根据以上各小节分析，在已建立的仿真模型基础上，用 MATLAB 语言编程实现。具体实现时应注意采取合理的方法。

（1）系统参数输入方法　为利用 MATLAB 语言的矩阵处理功能，系统参数输入按各环节 a_i，b_i，c_i，d_i 输入参数矩阵 $\boldsymbol{P}$。

如

$$\boldsymbol{P} = \begin{bmatrix} a_1, & b_1, & c_1, & d_1 \\ a_2, & b_2, & c_2, & d_2 \\ \cdots & \cdots & \cdots & \cdots \\ a_n, & b_n, & c_n, & d_n \end{bmatrix}$$

然后由程序自动形成式（3-6）中 $\boldsymbol{A}$、$\boldsymbol{B}$、$\boldsymbol{C}$、$\boldsymbol{D}$ 各阵和式（3-10）中闭环状态方程各系数阵。

（2）连接矩阵输入方法　连接阵 $\boldsymbol{W}$ 和 $\boldsymbol{W}_0$ 中大量的元素是零元素，而表示连接关系的 w_{ij} 非零元素并不多。因此，程序中采用只输入非零元素的方式，可大大加快输入速度，输入后由程序内部自动形成完整的连接矩阵 $\boldsymbol{W}$ 和 $\boldsymbol{W}_0$，具体方法如下：

建立非零元素矩阵 $\boldsymbol{W}_{IJ}$（$m\times 3$ 型），将非零元素按照 i，j，w_{ij} 次序逐行输入。其中，i 为被作用环节号；j 为作用环节号；w_{ij} 为作用关系值（连接系数）；m 为非零元素个数（包括 $\boldsymbol{W}_0$ 阵中非零元素）。

例如，按照图 3-12 和式（3-8），相应的 $\boldsymbol{W}_{IJ}$ 应按如下格式写出

$$\boldsymbol{W}_{IJ}=\begin{bmatrix}1 & 0 & 1\\ 1 & 3 & -v_3\\ 2 & 1 & 1\\ 2 & 2 & -v_2\\ 3 & 2 & 1\end{bmatrix}$$

可知各非零元素 w_{ij} 在第 3 列，其在 $\boldsymbol{W}$ 和 $\boldsymbol{W}_0$ 中的位置（即下标）由第 1 列、第 2 列元素值决定。而下标为 0，表示系统外加参考输入作用，相应 w_{i0} 为 $\boldsymbol{W}_0$ 中的元素。

（3）程序框图　框图如图 3-13 所示。框图中四阶龙格-库塔求解部分完全可调用上一节程序 sp3_1. m 中的运行程序块。

为方便起见，输入仍为阶跃函数，幅值由 y_0 值确定。为保证精度，又不致于输出太多点数，数值求解过程分两层循环，内层循环保证精度，步长可取小。外层每循环一次，输出数据一次，以便进行分析。

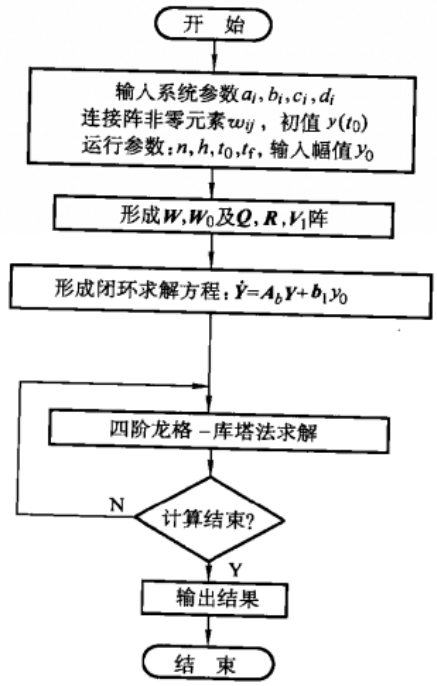


图 3-13　复杂连接闭环系统的仿真程序框图

（4）程序实现　仍采用 MATLAB 语言编程。

1）输入数据：

```
% ---------- filename:sp3_2.m 面向复杂连接闭环系统的仿真程序----------
P = [a1,b1,c1,d1;
     a2,b2,c2,d2;
     …  …  …  …;
     an,bn,cn,dn];          %各环节参数输入
WIJ = [i,j,wij; …  …];      %连接阵非零元素输入
n = n0;                     %环节个数(系统阶次)
Y0 = y0;                    %阶跃输入幅值
Yt0 = [y10,y20,…  …,yn0];   %各环节初值
h = h0;                     %计算步长
L1 = l1;                    %打印间隔点数(每隔点 l1 输出一次)
T0 = t0;
Tf = tf;
nout = nout;                %输出环节的编号
```

2）形成闭环各系数阵：

```
A = diag(P(:,1));B = diag(P(:,2));
C = diag(P(:,3));D = diag(P(:,4));
m = length(WIJ(:,1));                    % 求非零元素个数
W0 = zeros(n,1);W = zeros(n,n);          % 建立初始 W(n×n 维方阵)、W0 阵(n 维列向量)
    for k = 1: m
        if (WIJ(k,2) = =0);  W0(WIJ(k,1)) = WIJ(k,3);
        else  W(WIJ(k,1),WIJ(k,2)) = WIJ(k,3);
        end;
    end;                                 % 求 W0 阵和 W 阵
Q = B - D*W;      Qn = inv(Q);           % 求 Q 和 Q^-1 阵
R = C*W - A;      V1 = C*W0;             % 求 R 和 V1 阵
Ab = Qn*R;      b1 = Qn*V1;              % 形成闭环系数阵。
```

3）数值积分求解：

```
Y = Yt0';y = Y(nout);t = T0;             % 置初值,做好求解准备
N = round((Tf - T0)/(h*L1));             % 总输出点数
    for i = 1: N;                        % 每循环一次,输出一点数据
        for j = 1: L1;                   % 每输出点之间计算 L1 次
        %    (调用四阶龙格—库塔法程序块,求解 Ẏ = A_b Y + b_1 y_0)
                                         ⋮
                                         ⋮
        end;
        y = [y,Y(nout)];                 % 保存输出环节动态响应值
        t = [t,t(i) + h*L1];             % 保存时间向量
    end;
```

4）输出结果：

```
[t',y']          % 输出响应数据
plot(t,y)        % 输出响应图形
```

同样，在以上程序语句中用到了以下 MATLAB 语言提供的特殊功能函数：

diag（ ）；　　产生以括号内向量元素作为主对角元素的对角阵

P（：，I）；　　取 $\boldsymbol{P}$ 阵中第 I 列的所有元素作为单独列向量

若需要观察其他环节动态响应，可重新输入 n_{out} 值，再次运行程序，即可得到另一环节输出变化曲线、数据。

4. 计算步长的选择

前节及本节所述仿真程序均采用四阶龙格-库塔法求解状态方程，方法本身为四阶精度，误差与 h^5 同数量级，应该说精度是能满足一般工程实际要求的。但仍然应注意若计算步长选取不恰当，会造成数值稳定性差的问题。

程序中采用定步长计算方法，即计算步长 h_0 是固定不变的，这样计算过程简便，误差也在工程设计允许范围之内。

通常可按以下经验数据选择四阶龙格-库塔法的定步长值：

$$h_0 \leqslant \frac{1}{5\omega_c}, \quad \omega_c \text{ 为系统开环频率特性的剪切频率}$$

或　　$h_0 \leqslant \frac{t_r}{10}$，　　t_r 为系统阶跃响应的上升时间

$h_0 \leqslant \frac{t_s}{40}$，　　t_s 为系统阶跃响应的调节时间（过渡过程时间）

若系统中有局部闭环，则以上各值应按反应速度最快的局部闭环考虑。当 t_r、t_s 大致能估计范围时，以上经验数据意味着，为充分反映系统响应开始阶段变化较快的情况，在 t_r 内至少应计算 10 个点；或为全面反映系统响应整个过渡过程变化情况，在 t_s 内至少应计算 40 个点。

还应注意到，h_0 的选取应小于系统中最小时间常数 τ 的两倍，即

$$h_0 < 2\tau$$

以保证数值计算的稳定性，得到较可靠的结果。

第三节　环节的离散化与非线性系统的数字仿真

前面几节介绍的仿真方法都是针对线性连续控制系统而言的，无论如何变化，最终都是通过数值积分方法求得结果。采用数值积分方法进行系统仿真的特点，我们已在第二章讨论过，这里来分析一下其不足之处：

1）单步法求解过程中，每计算一个步长 h，要多次求取函数的导数值，以获得不同方法下的各次斜率 K_1、K_2 等，步骤繁琐。

2）多步法求解过程中，又要求存储各状态变量值前 r 次时刻的数据，系统阶次较高时，存储量相当大，而且启动时还需要其他算法配合。

3）隐式算法求解必须经若干次迭代，才取得一个时刻的变量数值，计算速度受影响。

4）虽能得到各线性环节的输出响应值，但由于数值积分方法本身的原因所限，它是统一由状态方程求解变量值，对单个环节输出的特殊变化（如非线性变化）难以单独考虑，故还不能对环节中含有非线性特性的情况进行仿真。

由此可知，在工程精度允许的条件下，应寻找其他方法求解系统状态方程，克服以上不足之处，尤其希望找到能灵活处理典型非线性系统问题的仿真方法，它更具有实际应用意义。

一、连续系统的离散化模型法

差分方程是描述离散（采样）控制系统的数学模型，其主要特点是方程中各变量由各相邻时刻的变化量（差分及高阶差分关系）制约，方程一旦列出，就相当于得到递推方程。从初始时刻开始，可以递推求出各离散时刻的状态变量值。因此，在一定条件下，将连续系统的微分方程形式的状态方程转化为差分方程形式的状态方程组，就可以达到不必采用数值积分求解的目的。

连续系统状态方程的离散化在控制理论课程中已有详细叙述。这里从数字仿真的角度简要重述如下：

（一）连续系统状态方程的解及其离散化

设连续系统状态方程为

$$\begin{cases}\dot{\boldsymbol{X}} = \boldsymbol{AX} + \boldsymbol{BU} \\ \boldsymbol{Y} = \boldsymbol{CX} + \boldsymbol{DU}\end{cases}$$

$\boldsymbol{X}(t_0) = \boldsymbol{X}_0$ 为状态初始值

则由现代控制理论基础可知，状态变量 $\boldsymbol{X}(t)$ 的解为

$$\boldsymbol{X}(t) = \boldsymbol{\Phi}(t)\boldsymbol{X}_0 + \int_0^t \boldsymbol{\Phi}(t-\tau)\boldsymbol{BU}(\tau)\mathrm{d}\tau$$

其中，$\boldsymbol{\Phi}(t)$ 为状态转移矩阵，当状态方程为线性定常时，$\boldsymbol{\Phi}(t)$ 为矩阵指数形式

$$\boldsymbol{\Phi}(t) = \mathrm{e}^{\boldsymbol{A}t} = \boldsymbol{I} + \boldsymbol{A}t + \frac{1}{2!}\boldsymbol{A}^2t^2 + \cdots + \frac{1}{n!}\boldsymbol{A}^nt^n + \cdots$$

或

$$\mathrm{e}^{\boldsymbol{A}t} = L^{-1}\{(s\boldsymbol{I} - \boldsymbol{A})^{-1}\}$$

于是

$$\boldsymbol{X}(t) = \mathrm{e}^{\boldsymbol{A}t}\boldsymbol{X}_0 + \int_0^t \mathrm{e}^{\boldsymbol{A}(t-\tau)}\boldsymbol{BU}(\tau)\mathrm{d}\tau$$

连续系统状态解中，当 $t = kT$ 时，上式成为

$$\boldsymbol{X}(kT) = \mathrm{e}^{\boldsymbol{A}kT}\boldsymbol{X}_0 + \int_0^{kT} \mathrm{e}^{\boldsymbol{A}(kT-\tau)}\boldsymbol{BU}(\tau)\mathrm{d}\tau$$

而 $t = (k+1)T$ 时，可表示为

$$\begin{aligned}\boldsymbol{X}((k+1)T) &= \mathrm{e}^{\boldsymbol{A}(k+1)T}\boldsymbol{X}_0 + \int_0^{(k+1)T} \mathrm{e}^{\boldsymbol{A}((k+1)T-\tau)}\boldsymbol{BU}(\tau)\mathrm{d}\tau \\ &= \mathrm{e}^{\boldsymbol{A}kT}\mathrm{e}^{\boldsymbol{A}T}\boldsymbol{X}_0 + \int_0^{kT} \mathrm{e}^{\boldsymbol{A}(kT-\tau)}\mathrm{e}^{\boldsymbol{A}T}\boldsymbol{BU}(\tau)\mathrm{d}\tau + \int_{kT}^{(k+1)T} \mathrm{e}^{\boldsymbol{A}((k+1)T-\tau)}\boldsymbol{BU}(\tau)\mathrm{d}\tau \\ &= \mathrm{e}^{\boldsymbol{A}T}\left[\mathrm{e}^{\boldsymbol{A}kT}\boldsymbol{X}_0 + \int_0^{kT} \mathrm{e}^{\boldsymbol{A}(kT-\tau)}\boldsymbol{BU}(\tau)\mathrm{d}\tau\right] + \int_{kT}^{(k+1)T} \mathrm{e}^{\boldsymbol{A}((k+1)T-\tau)}\boldsymbol{BU}(\tau)\mathrm{d}\tau \\ &= \mathrm{e}^{\boldsymbol{A}T}\boldsymbol{X}(kT) + \left[\int_0^T \mathrm{e}^{\boldsymbol{A}(T-\tau)}\boldsymbol{B}\mathrm{d}\tau\right]\boldsymbol{U}(kT)\end{aligned} \tag{3-11}$$

以上积分式中应用了 $t = \tau - kT$，$\tau = t + kT$，$\mathrm{d}\tau = \mathrm{d}t$ 变量代换，并认为在 kT 到 $(k+1)T$ 采样时刻内，$\boldsymbol{U}(\tau)$ 为常量，即保持为 $\boldsymbol{U}(kT)$ 值一周期内不变，相当于零阶保持器的作用。所以

$$\boldsymbol{X}((k+1)T) = \boldsymbol{\Phi}(T)\boldsymbol{X}(kT) + \boldsymbol{\Phi}_{\mathrm{m}}(T)\boldsymbol{U}(kT) \tag{3-12}$$

是典型的离散系统一阶差分方程组。

其中，$\boldsymbol{\Phi}(T) = \mathrm{e}^{\boldsymbol{A}T}$，为 $t = T$ 时的状态转移矩阵

$$\boldsymbol{\Phi}_{\mathrm{m}}(T) = \int_0^T \mathrm{e}^{\boldsymbol{A}(T-\tau)}\boldsymbol{B}\mathrm{d}\tau = \int_0^T \boldsymbol{\Phi}(T-\tau)\boldsymbol{B}\mathrm{d}\tau$$

式 (3-12) 即为系统离散化后的差分方程，由差分方程特点可知，该式实际上就是状态变量的数

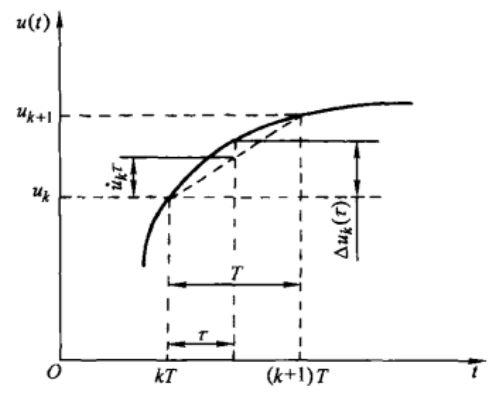


图 3-14 两采样时刻之间 $\boldsymbol{U}(\tau)$ 的变化

值求解递推公式，即

$$\boldsymbol{X}_{k+1}=\boldsymbol{\Phi}\ (T)\ \boldsymbol{X}_k+\boldsymbol{\Phi}_{\mathrm{m}}\ (T)\ \boldsymbol{U}_k \tag{3-13}$$

若希望递推公式精度更高些，应该考虑到在两次采样时刻 kT、$(k+1)T$ 之间 $\boldsymbol{u}(\tau)$一直在变化，用一阶保持器近似更为合理，如图 3-14 所示。

将 $\boldsymbol{U}(\tau)$表示为随 $\Delta\boldsymbol{U}_k(\tau)$变化的函数，得

$$\boldsymbol{U}(\tau)=\boldsymbol{U}(kT)+\Delta\boldsymbol{U}_k(\tau)$$

而 $\Delta\boldsymbol{U}_k(\tau)$又可用下式近似表达

$$\Delta\boldsymbol{U}_k(\tau)=\frac{\boldsymbol{U}((k+1)T)-\boldsymbol{U}(kT)}{T}\tau\approx\dot{\boldsymbol{U}}\ (kT)\tau\qquad(0<\tau<T)$$

于是代入式(3-11)中积分项重新推导得

$$\begin{aligned}\int_0^T \mathrm{e}^{\boldsymbol{A}(T-\tau)}\boldsymbol{B}\boldsymbol{U}(\tau)\mathrm{d}\tau&=\int_0^T \mathrm{e}^{\boldsymbol{A}(T-\tau)}\boldsymbol{B}[\boldsymbol{U}(kT)+\Delta\boldsymbol{U}_k(\tau)]\mathrm{d}\tau\\&=\int_0^T \mathrm{e}^{\boldsymbol{A}(T-\tau)}\boldsymbol{B}\boldsymbol{U}(kT)\mathrm{d}\tau+\int_0^T \mathrm{e}^{\boldsymbol{A}(T-\tau)}\boldsymbol{B}\dot{\boldsymbol{U}}(kT)\tau\mathrm{d}\tau\\&=\boldsymbol{\Phi}_{\mathrm{m}}(T)\boldsymbol{U}(kT)+\boldsymbol{\Phi}_{\mathrm{j}}(T)\dot{\boldsymbol{U}}(kT)\end{aligned}\tag{3-14}$$

显然

$$\boldsymbol{X}_{k+1}=\boldsymbol{\Phi}(T)\boldsymbol{X}_k+\boldsymbol{\Phi}_{\mathrm{m}}(T)\boldsymbol{U}_k+\boldsymbol{\Phi}_{\mathrm{j}}(T)\dot{\boldsymbol{U}}_k \tag{3-15}$$

其中，$\boldsymbol{\Phi}(T)$、$\boldsymbol{\Phi}_{\mathrm{m}}(T)$意义同前

$$\boldsymbol{\Phi}_{\mathrm{j}}(T)=\int_0^T \mathrm{e}^{\boldsymbol{A}(T-\tau)}\boldsymbol{B}\tau\mathrm{d}\tau=\int_0^T \boldsymbol{\Phi}(T-\tau)\boldsymbol{B}\tau\mathrm{d}\tau \tag{3-16}$$

$\boldsymbol{\Phi}(T)$、$\boldsymbol{\Phi}_{\mathrm{m}}(T)$、$\boldsymbol{\Phi}_{\mathrm{j}}\ (T)$ 统称连续系统的离散化系数矩阵。

利用式（3-15）就可以编程对某连续系统进行仿真运算了。事先应离线求取：

$$\begin{aligned}\boldsymbol{\Phi}\ (t)&=\mathrm{e}^{\boldsymbol{A}t}\\\boldsymbol{\Phi}_{\mathrm{m}}(t)&=\int_0^T \mathrm{e}^{\boldsymbol{A}(t-\tau)}\boldsymbol{B}\mathrm{d}\tau=\int_0^T \boldsymbol{\Phi}(t-\tau)\boldsymbol{B}\mathrm{d}\tau\\\boldsymbol{\Phi}_{\mathrm{j}}(t)&=\int_0^T \mathrm{e}^{\boldsymbol{A}(t-\tau)}\boldsymbol{B}\tau\mathrm{d}\tau=\int_0^T \boldsymbol{\Phi}(t-\tau)\boldsymbol{B}\tau\mathrm{d}\tau\end{aligned}$$

再令阵中 $t=T$，即得离散化矩阵 $\boldsymbol{\Phi}\ (T)$、$\boldsymbol{\Phi}_{\mathrm{m}}\ (T)$、$\boldsymbol{\Phi}_{\mathrm{j}}\ (T)$。

利用 MATLAB 语言控制系统工具箱中的功能函数，可使离散化系数矩阵 $\boldsymbol{\Phi}(T)$、$\boldsymbol{\Phi}_{\mathrm{m}}(T)$、$\boldsymbol{\Phi}_{\mathrm{j}}(T)$的求取过程大为简化。若已知连续系统状态方程各阵模型参数（$\boldsymbol{A}$、$\boldsymbol{B}$、$\boldsymbol{C}$、$\boldsymbol{D}$）以及采样周期 T，则语句

```
[F, G] =c2d (A, B, T);
```

返回的矩阵 F、G 就是所要求的 $\boldsymbol{\Phi}(T)$、$\boldsymbol{\Phi}_{\mathrm{m}}(T)$。

如果考虑精度高一些的、输入加了一阶保持器的算法，则在求得 F、G 后，再用一条组合语句

```
H = inv(A)^2 * (F - eye(n)) * B - inv(A) * B * T;
```

得到的矩阵 H 就是所要求的 $\boldsymbol{\Phi}_{\mathrm{j}}(T)$。语句中所用的求取公式为

$$\boldsymbol{\Phi}_{\mathrm{j}}(T)=\boldsymbol{A}^{-2}[\boldsymbol{\Phi}(T)-\boldsymbol{I}]\boldsymbol{B}-\boldsymbol{A}^{-1}\boldsymbol{B}T$$

是对式（3-16）通过分部积分得到的。

MATLAB 中还提供了功能更强的求取连续系统离散化矩阵的函数 c2dm()。它与 c2d () 的区别在于:允许使用者在调用时自行选择确定离散化变换方式（见表 3-2），并且所得到的是标准的离散化状态方程

$$\begin{cases} \boldsymbol{X}_{k+1} = \boldsymbol{Ad}\boldsymbol{X}_k + \boldsymbol{Bd}\boldsymbol{U}_k \\ \boldsymbol{Y}_{k+1} = \boldsymbol{Cd}\boldsymbol{X}_k + \boldsymbol{Dd}\boldsymbol{U}_k \end{cases}$$

式中的各系数阵（***Ad***，***Bd***，***Cd***，***Dd***）也可由语句求得。

语句调用格式如下:

```
[Ad, Bd, Cd, Dd] = c2dm (A, B, C, D, T,'选项');
```

与其他转换方式类似，语句

```
[A, B] = d2c(F, G, T);
[A, B, C, D] = d2cm (Ad, Bd, Cd, Dd, T,'选项');
```

是离散化过程的逆过程，它即是用以完成从离散化系统转换为连续系统各系数阵求取过程的功能函数。有关这些语句使用的详细情况请参阅书后有关参考文献。

表 3-2 离散化变换方式选项

选 项	说 明
'zoh'	假设对输入信号加一个零阶保持器
'foh'	假设对输入信号加一个一阶保持器
'tustin'	双线性变换方法（Tustin 算法）
'prewarp'	改进的 Tustin 变换方法
'matched'	SISO 系统的零极点匹配法

这种方法通常称为按系统离散化方法，只能对定常系统进行仿真，虽避开了数值积分求解过程，但由推导过程知，其精度要略差些，仅满足工程需要。该法仍未解决非线性环节的仿真问题，要在仿真程序中包含非线性环节，必须从结构图入手，对环节进行离散化，按环节离散化模型考虑问题。

（二）典型环节状态方程的离散化

对阶次较高的系统，系统离散化后求取 $\boldsymbol{\Phi}(T)$、$\boldsymbol{\Phi}_m(T)$和 $\boldsymbol{\Phi}_j(T)$并非易事，遇到复杂连接的系统结构图时，欲采用上述方法，事前需设法化简、合并，还要做大量的工作，这就失去了计算机仿真运算的意义。因此，下面考虑如何把典型环节连续模型化为离散模型，使离散化仿真模型也能面向复杂连接系统。

仍以式（3-5）定义的典型环节为标准，即

$$G_i(s) = \frac{y_i(s)}{u_i(s)} = \frac{C_i + D_i s}{A_i + B_i s} \qquad (i = 1, 2, \cdots, n)$$

为方便起见，略去下标 i，并改写为

$$G(s) = \frac{y(s)}{u(s)} = \frac{C}{A + Bs} + \frac{C}{A + Bs}\frac{D}{C}s \tag{3-17}$$

令

$$x = \frac{C}{A + Bs}u \tag{3-18}$$

则

$$y = \frac{C}{A + Bs}u + \frac{C}{A + Bs}u\frac{D}{C}s = x + \frac{D}{C}sx \tag{3-19}$$

显然，由式（3-18）、式（3-19）易得对应典型环节的连续微分方程

$$\begin{cases} \dot{x} = -\dfrac{A}{B}x + \dfrac{C}{B}u \\ y = \left(1 - \dfrac{AD}{CB}\right)x + \dfrac{D}{B}u \end{cases} \tag{3-20}$$

按离散化步骤，可推得

$$\begin{cases} x_{k+1} = \boldsymbol{\Phi}(T)x_k + \boldsymbol{\Phi}_{\mathrm{m}}(T)u_k + \boldsymbol{\Phi}_{\mathrm{j}}(T)\dot{u}_k \\ y_{k+1} = \boldsymbol{\Phi}_{\mathrm{c}}x_{k+1} + \boldsymbol{\Phi}_{\mathrm{d}}u_{k+1} \end{cases} \tag{3-21}$$

其中

$$\begin{cases} \boldsymbol{\Phi}(T) = \mathrm{e}^{-\frac{A}{B}T} \\ \boldsymbol{\Phi}_{\mathrm{m}}(T) = \displaystyle\int_0^T \mathrm{e}^{-\frac{A}{B}(T-\tau)} \dfrac{C}{B}\mathrm{d}\tau = \dfrac{C}{A}(1 - \mathrm{e}^{-\frac{A}{B}T}) \\ \boldsymbol{\Phi}_{\mathrm{j}}(T) = \displaystyle\int_0^T \mathrm{e}^{-\frac{A}{B}(T-\tau)} \dfrac{C}{B}\tau\mathrm{d}\tau = \dfrac{C}{A^2}(AT - B + B\mathrm{e}^{-\frac{A}{B}T}) \\ \boldsymbol{\Phi}_{\mathrm{c}} = 1 - \dfrac{AD}{CB} \\ \boldsymbol{\Phi}_{\mathrm{d}} = \dfrac{D}{B} \end{cases} \tag{3-22}$$

对典型一阶环节来说，它们均为标量系数，且求法固定。只要将已知 A、B、C、D 各值代入式(3-22)即可求得离散化系数 $\boldsymbol{\Phi}(T)$、$\boldsymbol{\Phi}_{\mathrm{m}}(T)$、$\boldsymbol{\Phi}_{\mathrm{j}}(T)$以及 $\boldsymbol{\Phi}_{\mathrm{c}}$、$\boldsymbol{\Phi}_{\mathrm{d}}$。

例如：① 积分环节：$G(s) = \dfrac{K}{s}$，$A=0$，$B=1$，$C=K$，$D=0$，则

$$\begin{cases} \boldsymbol{\Phi}(T) = \mathrm{e}^{-\frac{A}{B}T} = 1 \\ \boldsymbol{\Phi}_{\mathrm{m}}(T) = \lim\limits_{A\to 0}\dfrac{C}{A}(1 - \mathrm{e}^{-\frac{A}{B}T}) = KT \\ \boldsymbol{\Phi}_{\mathrm{j}}(T) = \lim\limits_{A\to 0}\dfrac{C}{A^2}(AT - B + B\mathrm{e}^{-\frac{A}{B}T}) = \dfrac{KT^2}{2} \\ \boldsymbol{\Phi}_{\mathrm{c}} = 1 \qquad \boldsymbol{\Phi}_{\mathrm{d}} = 0 \end{cases}$$

所以，状态与输出递推公式为

$$\begin{cases} x_{k+1} = x_k + KTu_k + \dfrac{KT^2}{2}\dot{u}_k \\ y_{k+1} = x_{k+1} \end{cases}$$

② 积分比例环节：$G(s) = \dfrac{K(bs+1)}{s}$，$A=0$，$B=1$，$C=K$，$D=bK$。

由于 A、B、C 均与①相同，故 $\boldsymbol{\Phi}(T)$、$\boldsymbol{\Phi}_{\mathrm{m}}(T)$、$\boldsymbol{\Phi}_{\mathrm{j}}(T)$和 $\boldsymbol{\Phi}_{\mathrm{c}}$ 与①完全相同，相应状态方程也完全相同。

但因 $D\neq 0$，只有 $\boldsymbol{\Phi}_{\mathrm{d}}$ 不同，所以应注意输出方程成为

$$y_{k+1} = x_{k+1} + Kbu_{k+1}$$

③ 惯性环节：$G(s)=\dfrac{K}{s+a}$，$A=a$，$B=1$，$C=K$，$D=0$，则

$$\begin{cases}\boldsymbol{\Phi}(T)=\mathrm{e}^{-aT}\\ \boldsymbol{\Phi}_{\mathrm{m}}(T)=\dfrac{K}{a}(1-\mathrm{e}^{-aT})\\ \boldsymbol{\Phi}_{\mathrm{j}}(T)=\dfrac{K}{a^2}(aT-1+\mathrm{e}^{-aT})\\ \boldsymbol{\Phi}_{\mathrm{c}}=1 \qquad \boldsymbol{\Phi}_{\mathrm{d}}=0\end{cases}$$

状态与输出递推公式为

$$\begin{cases}x_{k+1}=\mathrm{e}^{-aT}x_k+\dfrac{K}{a}(1-\mathrm{e}^{-aT})u_k+\dfrac{K}{a^2}(aT-1+\mathrm{e}^{-aT})\dot{u}_k\\ y_{k+1}=x_{k+1}\end{cases}$$

④ 比例惯性环节 $K\dfrac{s+b}{s+a}=b\dfrac{\frac{K}{b}s+K}{s+a}$，$A=a$，$B=1$，$C=K$，$D=\dfrac{K}{b}$。

很明显，因为 A、B、C 均与③相同，状态方程离散化系数 $\boldsymbol{\Phi}(T)$、$\boldsymbol{\Phi}_{\mathrm{m}}(T)$、$\boldsymbol{\Phi}_{\mathrm{j}}(T)$ 也相同，故状态方程递推公式与③完全相同。又因为 $D\neq 0$，$\boldsymbol{\Phi}_{\mathrm{c}}$、$\boldsymbol{\Phi}_{\mathrm{d}}$ 与③不同，故输出方程需特别考虑。

注意到 $\left(\dfrac{y}{b}\right)=\dfrac{\frac{K}{b}s+K}{s+a}u$ 时，对应有

$$\left(\frac{y}{b}\right)=\left(1-\frac{AD}{CB}\right)x+\frac{D}{B}u$$

容易推得 $$y_{k+1}=(b-a)x_{k+1}+Ku_{k+1}$$

显然 $$\boldsymbol{\Phi}_{\mathrm{c}}=b-a,\qquad \boldsymbol{\Phi}_{\mathrm{d}}=K$$

由上可知，四种典型环节包括了一阶环节的所有不同形式，其离散化系数能统一由 K、a、b 三个参数表达，它们之间的相互关系列为表 3-3。

表 3-3 离散化环节参数表

	K/s	$K(1+bs)/s$	$K/(s+a)$	$K(s+b)/(s+a)$
$\boldsymbol{\Phi}$	1		e^{-aT}	
$\boldsymbol{\Phi}_{\mathrm{m}}$	KT		$(K/a)(1-\mathrm{e}^{-aT})$	
$\boldsymbol{\Phi}_{\mathrm{j}}$	$KT^2/2$		$(K/a^2)[aT-(1-\mathrm{e}^{-aT})]$	
$\boldsymbol{\Phi}_{\mathrm{c}}$	1		1	$b-a$
$\boldsymbol{\Phi}_{\mathrm{d}}$	0	Kb	0	K
K	C/B	C/B	C/B	D/B
a	0	0	A/B	A/B
b	0	D/C	0	C/D

这种方法通常称为按环节离散化方法，其精度有限，由于典型环节离散化模型的状态递推方程包含 $\dot{u}_k$ 项，故环节仿真精度仅为二阶，因此整个系统仿真精度也不会高于二阶。但该法的突出优点是可以在仿真过程中考虑非线性环节影响，从而使对各类系统仿真的适应能力得到增强。

（三）按环节离散化数字仿真程序的实现

把以上原理用计算机仿真程序实现出来，称为连续系统按环节离散化仿真，有的参考文献中也称之为离散相似法。实现过程中要解决好以下几个问题：

1. 典型环节的转化

前节所述四种环节的离散化系数 $\boldsymbol{\Phi}(T)$、$\boldsymbol{\Phi}_m(T)$、$\boldsymbol{\Phi}_j(T)$以及 $\boldsymbol{\Phi}_c$、$\boldsymbol{\Phi}_d$ 的求取有固定规律可循，所得结果又能方便地递推求解，故程序中主要应解决好如何把任意典型一阶环节转化为这四种离散化标准环节形式，并尽可能采用一致的方法求取各环节离散化系数，仿真过程中最好按统一公式求对应环节的状态值 x 和输出值 y，充分利用 MATLAB 语言的特点，使程序功能强却不繁琐、冗长。其思路如下：

典型环节数据输入后，首先判断 A 是否为 0，即可分出①、②和③、④两组，这两组对应的状态方程离散化系数 $\boldsymbol{\Phi}(T)$、$\boldsymbol{\Phi}_m(T)$ 和 $\boldsymbol{\Phi}_j(T)$ 求取方法各自相同，可以直接套用相同求解公式求取后存入相应单元。但由于对应输出方程各有不同，故又需判断 D 是否为 0，以使输出方程离散化系数 $\boldsymbol{\Phi}_c$、$\boldsymbol{\Phi}_d$ 修正后，也存入相应单元。

求得各环节离散化系数后，结果存入相应数组单元 FI(I)、FIM(I)、FIJ(I)、FIC(I)以及 FID(I)，其中：I 表示环节序号。仿真运行时从各环节相应单元取出，分别求取各环节状态与输出即可。

据以上思路给出仿真程序框图如图 3-15 所示。

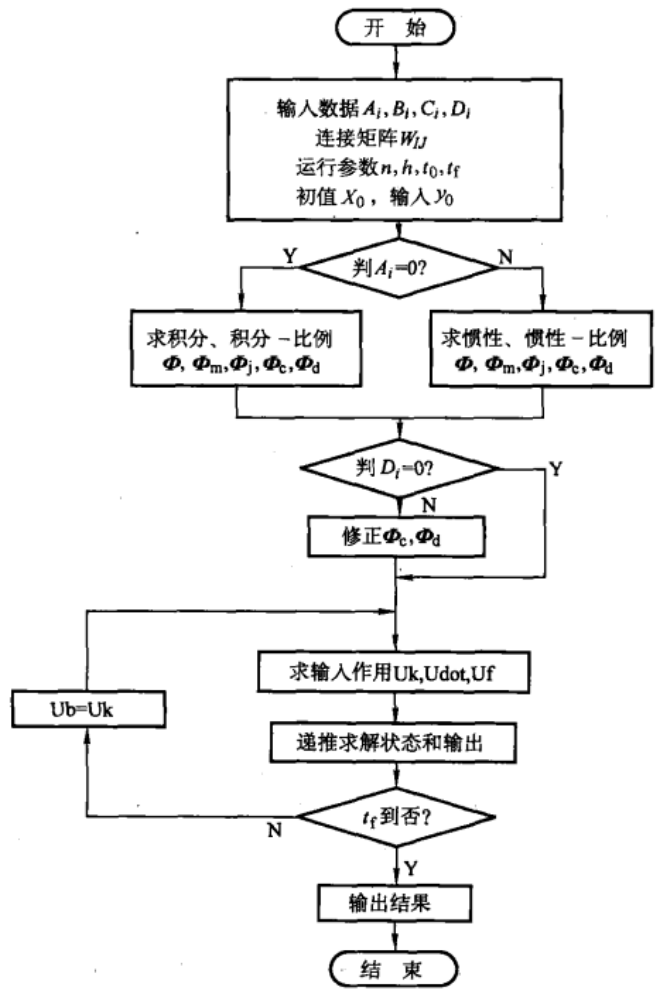


图 3-15　按环节离散化的仿真程序框图

2. 各环节输入作用

求 $t=(k+1)T$ 时刻的各环节状态 $\boldsymbol{X}_{k+1}$的递推计算式中要用到 u_k、$\dot{u}_k$。而求 $t=(k+1)T$ 时刻的各环节输出 $\boldsymbol{Y}_{k+1}$的递推计算式中还要用到 u_{k+1}。因此有必要讨论一下这几种输入作用形式的计算方法。

1）u_k 可通过连接矩阵直接求得，即

$$\boldsymbol{U}_k = \boldsymbol{W}\boldsymbol{Y}_k + \boldsymbol{W}_0 y_0 \qquad (3\text{-}23)$$

式中，$\boldsymbol{U}_k = [u_{1k}, u_{2k}, \cdots, u_{nk}]^T$ 为各环节输入量；n 为系统环节数，亦即系统阶数；$\boldsymbol{Y}_k = [y_{1k}, y_{2k}, \cdots, y_{nk}]^T$ 为各环节输出量；y_0 为外加参考输入量，为方便起见，这里仅考虑单变量阶跃输入情况。

2）$\dot{u}_k$ 利用近似表达式求取。即由已知的各环节输出值 $\boldsymbol{Y}_k$，按式（3-23）求出当前 $\boldsymbol{U}_k$

值，并取出所保存的前一步 U_{k-1} 值，按照下式即可求得 $\dot{U}_k$ 的近似值

$$\dot{U}_k \approx \frac{U_k - U_{k-1}}{h} \tag{3-24}$$

3）u_{k+1} 利用上面已求得的 U_k、$\dot{U}_k$ 在一个步长 h 内按一阶保持近似关系求取，即所谓折线近似法（见图 3-16）。由图 3-16 显然有

$$U_{k+1} = U_k + \Delta U_k = U_k + h\dot{U}_k$$

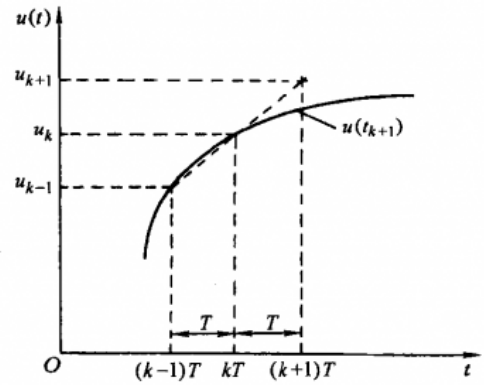


图 3-16 折线近似法求 u_{k+1}

3. 仿真程序实现

filename：sp3_3. m

1）输入数据及连接矩阵的产生与第二节程序 sp3_2. m 相同，注意这里输入幅值为 y_0（程序中用 Y0 表示）。

2）求离散化各系数：

```
%---------- filename:sp3_3.m 按环节离散化数字仿真程序----------
for i=1: n
    if(A(i)==0);                                    %求积分或积分比例环节各系数
        FI(i)=1;
        FIM(i)=h*C(i)/B(i);
        FIJ(i)=h*h*C(i)/B(i)/2;
        FIC(i)=1;FID(i)=0;                          %积分环节各系数求取完毕
        if(D(i)~=0);                                %若为积分比例,修正 Φd
            FID(i)=D(i)/B(i);
        else
        end
else                                                %求惯性或惯性比例环节各系数
    FI(i)=exp(-h*A(i)/B(i));
        FIM(i)=(1-FI(i))*C(i)/A(i);
        FIJ(i)=h*C(i)/A(i)-FIM(i)*B(i)/A(i);
        FIC(i)=1;FID(i)=0;                          %惯性环节各系数求取完毕
        if(D(i)≈=0);                                %若为惯性比例,修正 Φc、Φd
            FIM=(1-FI(i))*D(i)/A(i)
            FIJ(i)=h*D(i)/A(i)-FIM(i)*B(i)/A(i)
            FIC(i)=C(i)/D(i)-A(i)/B(i);
            FID(i)=D(i)/B(i);
        else
        end
        end
        end
```

3）求输入作用：

```
Y = zeros(n,1);X = Y;y = 0;Uk = zeros(n,1);Ub = Uk;    % 置初值
t = T0:h * L1:Tf;N = length(t);                        % 建立时间序列向量,N 维
    for k = 1:N - 1
      for l = 1:L1
          Ub = Uk;                                     % 保存前一次输入值
          Uk = W * Y + W0 * Y0;                        % 求当前 Uk,即 u_k
          Udot = (Uk - Ub)/h;                          % 求当前 Uk 的导数,即 u_k^·
          Uf = 2 * Uk - Ub;                            % 求下一步输入 Uf,即 u_{k+1}
```

4）求各环节状态及输出：

```
      X = FI'. * X + FIM'. * Uk + FIJ'. * Udot;        % 点运算求取状态和输出
      Y = FIC'. * X + FID'. * Uf;
end
```

5）输出结果：

```
        y = [y,Y(nout)];
    end
plot(t,y)
```

上面程序中又用到了几个 MATLAB 语言中的特殊运算符号：

:——冒号符，可用于生成行向量。如程序中

t = T0:h * L1:Tf

表示生成的 t = $[t_0, t_1, \cdots, t_N]$ 为一个 T0 到 Tf、间隔为 L1 倍步长 h、$N+1$ 维的行向量，即其中的 t_0 = T0，$t_1 = t_0 + h * L1, \cdots, t_N = (t_{N-1} + h * L1) \leqslant Tf$；

= =——逻辑等号符，MATLAB 中规定写法；

~ =——逻辑不等号符，MATLAB 中规定写法；

· *——点乘运算符，表示两同维矩阵中各相应位置上的元素进行相乘运算；类似地，MATLAB 还提供了其他点运算，如 ./和 .∧分别表示点除和点乘幂运算。

灵活运用这些功能，可使程序编制的简练、实用。更多的编程技巧请参阅 MATLAB 语言程序设计的有关文献资料。

4. 连续系统离散化数字仿真的特点

由本节讨论我们清楚地了解到连续系统离散化数字仿真的特点：

1）将连续系统离散化后进行仿真，可以用得到的离散方程递推求解状态和输出，避免了数值积分方法中繁琐的龙格-库塔系数（导函数）的求取过程，计算简便。

2）按环节离散化仿真，每步都可求出各环节输入、输出，很容易推广用以解决非线性系统仿真问题。

3）离散化过程中对原连续系统引入了虚拟采样开关和零阶（一阶）保持器，造成的滞后使得仿真计算误差增大，严重时甚至会引起系统数值计算不稳定。

按照香农（Shannon）定理，只有当采样周期 T 较小，采样频率 ω_s 满足大于 $2\omega_m$（系统最大截止频率，亦称有限频率带宽）时，离散化仿真才不会造成严重失真，能较好地反映

连续系统的动态响应性能。所以，此法应用时，计算步长 h 必须比数值积分方法选取得更小。一般有以下经验公式

$$h_0 \leqslant \frac{1}{50\omega_c}, \quad \omega_c \text{ 为系统开环频率特性的剪切频率}$$

或

$$h_0 \leqslant \frac{t_r}{100}, \quad t_r \text{ 为系统阶跃响应的上升时间}$$

$$h_0 \leqslant \frac{t_s}{400}, \quad t_s \text{ 为系统阶跃响应的调节时间（过渡过程时间）}$$

可见，与数值积分法相比，h 约小了一个数量级。

二、非线性系统的数字仿真

若系统中含有非线性元件，则给仿真模型的建立、计算都带来一些新问题。

非线性系统也可表示为状态方程描述形式（如式（2-78）），重写如下

$$\dot{\boldsymbol{X}}(t) = F(\boldsymbol{X}, \boldsymbol{U}, t), \quad \boldsymbol{X}(t_0) = \boldsymbol{X}_0$$

由式中可见，状态方程右端不再是状态变量 $\boldsymbol{X}$ 和输入函数 $\boldsymbol{U}$ 的线性组合，而是与 $\boldsymbol{X}$、$\boldsymbol{U}$ 有关的变量矩阵。这对采用前述仿真程序进行运算将造成困难，因为程序不能通用，所以对不同的系统仿真必须输入不同的自定义函数来表征非线性环节，且阶次越高的系统，越复杂、繁琐。至今为止，还没有统一的方法能解决所有的非线性问题。但对于一些包含常见典型非线性环节的控制系统，通过合理地建立模型、描述性能，利用上节讨论的按环节离散化仿真的方法，可以很方便地在程序中加入非线性环节，从而达到对非线性系统进行数字仿真的目的。

下面先来建立一些常见非线性环节的仿真模型。

（一）典型非线性环节及其仿真实现

1. 饱和非线性

图 3-17 所示饱和非线性环节的数学描述为

$$u_c = \begin{cases} -s_1 & u_r \leqslant -s_1 \\ u_r & -s_1 < u_r < s_1 \\ s_1 & u_r \geqslant s_1 \end{cases}$$

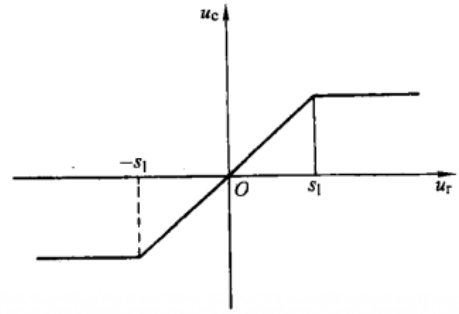


图 3-17 饱和非线性环节

相应仿真框图如图 3-18 所示。

据此，编制 MATLAB 程序如下：

```
% ---------- filename:satu.m 饱和非线性环节仿真程序----------
function Uc = satu(Ur,S1)
        if(abs(Ur) >= S1)
                if(Ur>0)
                        Uc = S1;
                else    Uc = -S1;
                end
        else    Uc = Ur;
        end
```

2. 死区非线性

图 3-19 所示死区非线性环节的数学描述为

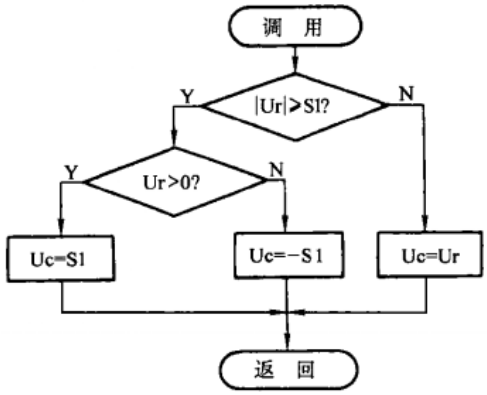


图 3-18　饱和非线性环节仿真框图

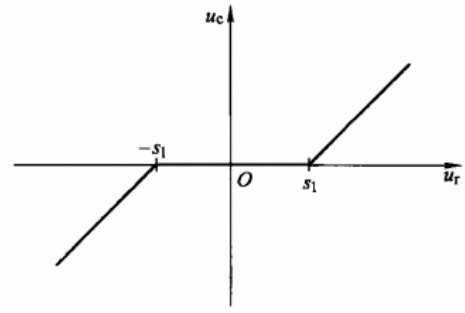


图 3-19　死区非线性环节

$$u_c=\begin{cases}u_r+s_1 & u_r\leqslant -s_1\\ 0 & -s_1<u_r<s_1\\ u_r-s_1 & u_r\geqslant s_1\end{cases}$$

相应仿真框图如图 3-20 所示。

据此，也可编制 MATLAB 程序如下：

```
%---------- filename:dead.m 死区非线性环节仿真程序----------
function Uc = dead(Ur,S1)
        if (abs(Ur) > =S1)
                if (Ur>0)
                        Uc = Ur - S1;
                else    Uc = Ur + S1;
                end
        else        Uc = 0;
        end
```

3. 滞环非线性

图 3-21 所示滞环非线性环节的数学描述为

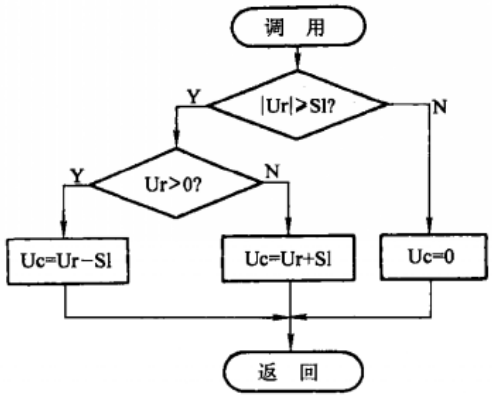


图 3-20　死区非线性环节仿真框图

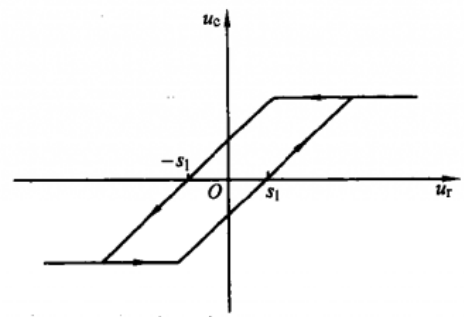


图 3-21　滞环非线性环节

$$u_c=\begin{cases}u_r-s_1 & \dot{u}_r>0 \text{ 且 } \dot{u}_c>0\\ u_r+s_1 & \dot{u}_r<0 \text{ 且 } \dot{u}_c<0\\ u_{cb} & \dot{u}_r<0 \text{ 且 } \dot{u}_c=0\\ u_{cb} & \dot{u}_r>0 \text{ 且 } \dot{u}_c=0\end{cases}$$

其中，u_{cb}表示非线性环节前一次的输出值。

从式中知，要正确地给 u_c 赋值，需要判断 u_r、u_c 的变化率大小。而这要通过与所存储的 u_r、u_c 的前一点值 u_{rb}、u_{cb}相比较来实现。即，若有

$$u_r-u_{rb}>0$$

则认为 $\dot{u}_r>0$，再由 (u_c-u_{cb}) 决定输出是取直线 $u_c=u_r-s_1$ 值，还是保持前次值 u_{cb}。而 $u_r-u_{rb}<0$ 情况读者可自行分析，方法与上类似。相应仿真框图如图 3-22 所示。

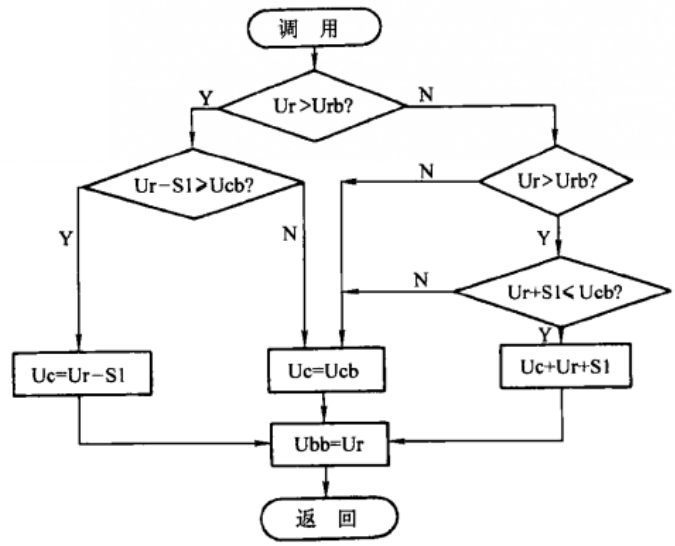


图 3-22 滞环非线性环节仿真框图

根据框图，编制 MATLAB 程序如下：

```
% ---------- filename:backlash.m 滞环非线性环节仿真程序----------
function [Uc,Ubb] = backlash (Urb,Ur,Ucb,Sl)
        if (Ur > Urb)                          % 输入 u_r 增大
            if((Ur - Sl) > = Ucb)              % 输出 u_c 也增大
                Uc = Ur - Sl;                  % 输出 u_c 取直线 u_r - s_1 上值
            else  Uc = Ucb;                    % 输出 u_c 没有增大,则保持前次值 u_cb
            end
        else  if(Ur < Urb)                     % 输入 u_r 减小
                if((Ur + Sl) < = Ucb)          % 输出 u_c 也减小
                    Uc = Ur + Sl;              % 输出 u_c 取直线 u_r + s_1 上值
                else  Uc = Ucb;                % 输出 u_c 没有减小,则保持前次值 u_cb
                end
            else  Uc = Ucb;                    % 输入没有变化,输出也保持不变
            end
        end
Ubb = Ur;                                      % 保留输入值作下次运算用
```

4. 继电非线性

图 3-23 所示继电非线性环节的数学描述为

$$u_c=\begin{cases}s_1 & u_r>0\\ -s_1 & u_r<0\end{cases}$$

相应框图及程序都较简单，读者可以自行画出、编程。

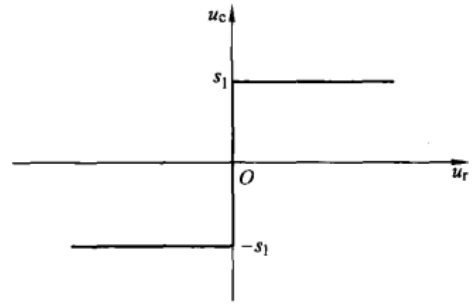


图 3-23　继电非线性环节

以上几种非线性环节的共同特点是都只需一个参数 s_1 就能反映出该环节的非线性特性，表达非常简练。如饱和环节中 s_1 就表示环节的饱和值；死区环节中 s_1 却表示环节的死区值；而在滞环环节中 s_1 又表示环节的滞环宽度值。不过要注意到，各环节的放大系数规定均为 1，若不为 1 则设法合入其前后的线性环节中，这样做也是为了建立模型方便的需要，而且在实际中并不困难。

5. 其他非线性

常见的非线性环节还有图 3-24 ~ 图 3-27 所示的一些形式，这些非线性环节比前几种要相对复杂些，但只要正确列出数学模型表达式，尽可能用最少的参数表出其特性，就不难画出程序框图、编制仿真程序并添加到主程序中去。

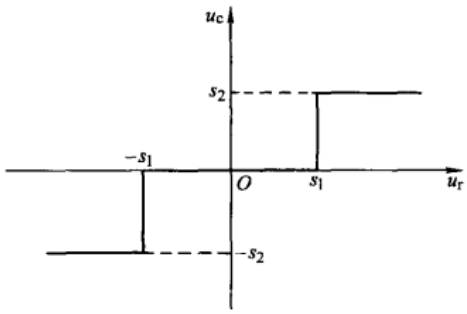


图 3-24　带死区三位继电非线性环节

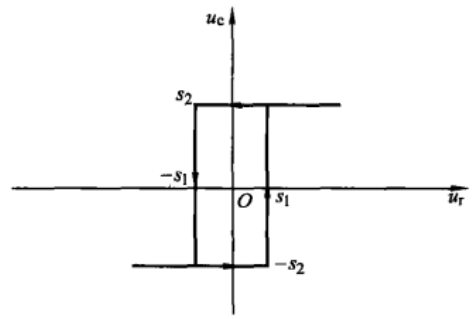


图 3-25　带滞环三位继电非线性环节

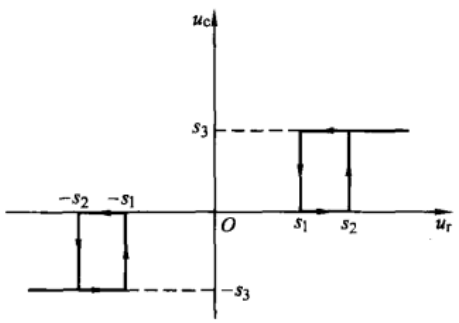


图 3-26　带死区滞环复杂继电非线性环节

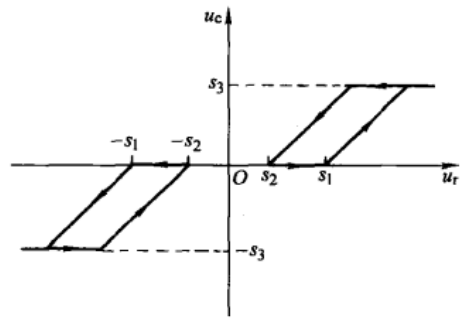


图 3-27　带死区的复杂滞环非线性环节

（二）非线性特征的判断

反映非线性环节特性的仿真子程序模块编制好后，还应设法将它们与仿真主程序有机地联系起来，使得在仿真分析系统的过程中能方便地调用它们。

利用按环节离散化的仿真程序，在输入数据时，设立非线性标志向量

$$\boldsymbol{Z}=[z_1,\ z_2,\ \cdots,\ z_n]$$

其中每一个元素对应标志一个环节的非线性特征，n 为环节序号。本程序中为说明问题方便，只考虑饱和、死区、滞环三种典型非线性特性，并规定非线性环节不单独作为一个环节进行运算，根据系统结构情况，把相应的非线性特性添加在某线性环节之前或之后。为此约定标志如下：

$z_i=0$	线性环节
$z_i=1$	线性环节前有饱和非线性，应修正 $U(i)$
$z_i=2$	线性环节前有死区非线性，应修正 $U(i)$
$z_i=3$	线性环节前有滞环非线性，应修正 $U(i)$
$z_i=4$	线性环节后有饱和非线性，应修正 $Y(i)$
$z_i=5$	线性环节后有死区非线性，应修正 $Y(i)$
$z_i=6$	线性环节后有滞环非线性，应修正 $Y(i)$

再设立一个参数向量单元 $\boldsymbol{S}$，专门用来存储非线性环节的参数 s_i。于是程序功能稍加扩展，就可用来对含有典型非线性环节的系统进行仿真分析了。

（三）程序功能扩展

在按环节离散化系统的仿真程序中，将以上方法作为扩展功能加入到适当位置，就可以方便地分析和研究非线性系统了。

1）求得 u_k 后，由 z_i 判断各环节入口有否非线性。若有，则根据标志值确定类型，转相应处理程序，修正 u_k 值。

2）求得 y_{k+1} 后，再由 z_i 判断各环节出口有否非线性。若有，则根据标志值确定类型，转相应处理程序，修正 y_{k+1} 值。

3）各种非线性特性按前节给出的程序，自定义为函数形式，以函数文件格式存储起来，由主程序在运行时调用。

（四）程序框图及仿真实现

程序框图如图 3-28 所示。

MATLAB 程序清单如下：

filename：sp3_4. m

1）数据输入及连接矩阵的形成均与程序 sp3_2. m 相同，同样应注意本程序中输入幅值要用 Y0 表达。

2）各环节离散系数求取与程序 sp3_3. m 相同。

3）求解运行部分程序为

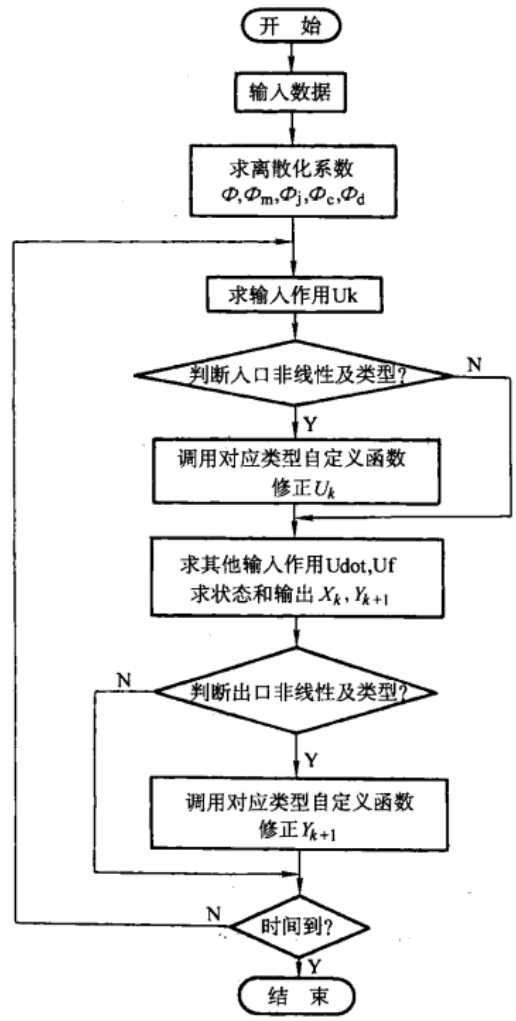


图 3-28 非线性系统的仿真程序框图

```
% ---------- filename:sp3_4.m 非线性系统的仿真程序----------
Y = zeros(n,l);X = Y;y = 0;Uk = zeros(n,l);Ubb = Uk;          % 设立一个向量单元 Ubb 存储非线性
                                                              % 环节输出的前一时刻值 ucb
t = T0:h*Ll:Tf;N = length(t);
for k = 1:N-1
    for l = l:Ll
        Ub = Uk;                                              % 多设立一个向量单元 Ub 存储非线性
                                                              % 环节输入 uk 前一时刻值
    Uk = W*Y + W0*Y0;                                         % 求当前 uk 值
    for i = l:n
        if(Z(i) ~ =0)                                         % 判各环节入口有否非线性
          if(Z(i) = =1)                                       % 若是饱和,调用饱和自定义函数
            Uk(i) = satu(Uk(i),S(i));
          end
          if(Z(i) = =2)                                       % 若是死区,调用死区自定义函数
            Uk(i) = dead(Uk(i),S(i));
          end
          if(Z(i) = =3)                                       % 若是滞环,调用滞环自定义函数
          [Uk(i),Ubb(i)] = backlash(Ubb(i),Uk(i),Ub(i),S(i));
          end
        end                                                   % 入口无非线性环节
      end                                                     % n 个环节入口判断完毕
      Udot = (Uk - Ub)/h;                                     % 求输入作用
      Uf = 2*Uk - Ub;
      X = FI'.*X + FIM'.*Uk + FIJ'.*Udot;                     % 求各环节状态
      Yb = Y;                                                 % 设立 Yb 向量单元存储 yk 前一时刻值
      Y = FIC'.*X + FID'.*Uf;                                 % 求当前 yk 值
      for i = l:n
          if(Z(i) ~ =0)                                       % 判各环节出口有否非线性
            if(Z(i) = =4)                                     % 若是饱和,调用饱和自定义函数
              Y(i) = satu(Y(i),S(i));
            end
            if(Z(i) = =5)                                     % 若是死区,调用死区自定义函数
              Y(i) = dead(Y(i),S(i));
            end
            if(Z(i) = =6)                                     % 若是滞环,调用滞环自定义函数
              [Y(i),Ubb(i)] = backlash(Ubb(i),Y(i),Yb(i),S(i));
            end
          end                                                 % 出口无非线性环节
        end                                                   % n 个环节出口判断完毕
      end                                                     % 时间间隔到
      y = [y,Y(nout)];                                        % 存储系统输出环节各时刻值
    end                                                       % 计算时间到
```

4）输出结果（同前）。

例 3-2 控制系统如图 3-29 所示，设输入阶跃函数幅值 $Y_0=10$，滞环非线性参数 $s_4=1$（滞环宽度）。1）不考虑非线性环节影响时，求解 $y(t)$ 的阶跃响应；2）考虑非线性环节影响，其余参数不变，求解 $y(t)$ 并与线性情况所得结果进行比较。

解 1）不考虑非线性环节影响时，求解过程如下：

① 先将环节编号标入图 3-29 中。

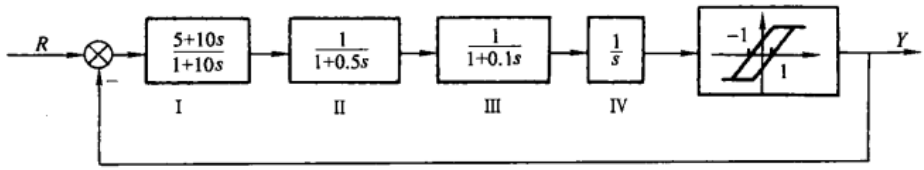


图 3-29 例 3-2 控制系统结构图

② 在 MATLAB 命令窗口下（以下语句前符号“>>”即表示 MATLAB 命令窗口环境），按编号依次将环节参数输入 $\boldsymbol{P}$ 阵，由图很显然

```
>> P = [1  10  5  10; 1  0.5  1  0; 1  0.1  1  0; 0  1  1  0];
```

③ 按各环节相对位置和连接关系，有连接矩阵如下

$$\boldsymbol{W}=\begin{bmatrix}0&0&0&-1\\1&0&0&0\\0&1&0&0\\0&0&1&0\end{bmatrix};\quad \boldsymbol{W}_0=\begin{bmatrix}1\\0\\0\\0\end{bmatrix}$$

程序中只需输入非零元素 w_{ij}，即按

$$\begin{matrix}\quad i & j & w_{ij}\end{matrix}$$
$$\boldsymbol{W}_{ij}=\begin{bmatrix}1&0&1\\2&1&1\\3&2&1\\4&3&1\\1&4&-1\end{bmatrix}$$

输入即可。

```
>> WIJ = [1  0  1; 2  1  1; 3  2  1; 4  3  1; 1  4  -1];
```

④ 各环节初始值为

```
>> X0 = [0  0  0  0];
```

⑤ 由于不考虑非线性影响，则非线性标志向量和参数向量均应赋零值；

```
>> Z = [0  0  0  0]; S = [0  0  0  0];
```

⑥ 输入运行参数；开环截止频率 ω_c 约为 1，故计算步长 h 取经验公式值，即 $h\leqslant 1/(50\omega_c)=0.02$，取 $h=0.01$；每 0.25s 输出一点，故取 $L_1=25$。

```
>> h = 0.01; L1 = 25;
>> n = 4;
>> T0 = 0;
>> Tf = 10;
>> nout = 4;
>> Y0 = 10;
```

⑦　运行 sp3_4.m，求解环节 IV 输出的 y_4 数值解的数据和响应曲线，如图 3-30 所示。

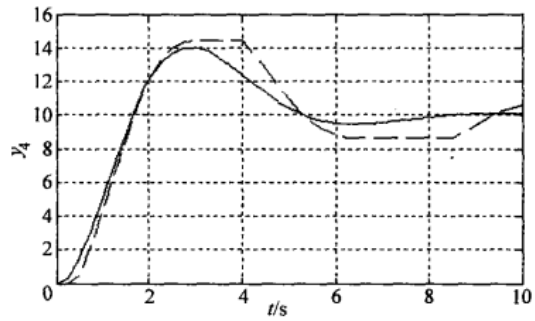


图 3-30　例 3-2 控制系统输出 y_4 响应曲线

```
>> sp3_4
```

2）考虑非线性环节 N 影响时，只需将非线性标志向量 $\boldsymbol{Z}$、参数向量 $\boldsymbol{S}$ 的相应分量正确输入即可。其余参数均保持不变，以便与线性情况比较。

①　由图 3-29 知，N 为环节 IV 之后的滞环非线性，滞环宽度 $s_4 = 1$。于是

```
>> Z = [0  0  0  6]; S = [0  0  0  1];
```

②　运行 sp3_4.m，仍求取 y_4 输出数据和响应曲线，如图 3-30 中虚线所示。

```
>> sp3_4
```

仿真结果清楚地表明了滞环非线性环节 N 对线性系统输出响应的影响。

综上可见，通过运用已掌握的仿真方法，又有 MATLAB 这样功能丰富、方便齐全的仿真软件作为手段，对常见的含有典型非线性环节的、复杂连接的控制系统仿真问题就能够很快得到所需的仿真实验结果。

第四节　计算机控制系统的数字仿真

计算机控制系统一般是指对被控制对象性能指标的控制由计算机或其他数字控制装置来完成的系统。因为这一类系统的特点是控制量不随时间连续变化，而是以一定的时间周期断续变化，如对系统误差量的检测，是按周期 T 定时采样并通过 A/D 单元变为数字信号进行处理的，这与传统意义上的采样控制系统完全类似，故也沿用采样控制系统的称法，或称数字控制系统。又因这类系统中的变量仅在离散时刻 $t = kT$ 有意义，在离散时刻之间是不定的，故也可称其为离散控制系统。

典型的计算机控制系统基本结构如图 3-31 所示。图中，A/D 单元相当于采样开关，把

模拟信号 $e(t)$ 转换为采样信号 $e(kT)$，其采样周期为 T；控制单元为计算机或其他数字控制器，将误差信号进行处理后送出控制信号；D/A 单元相当于采样保持器，把控制单元输出的数字信号 $u(kT)$ 转换为阶梯信号 $u_h(t)$。

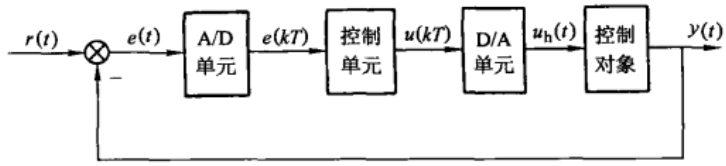


图 3-31 计算机控制系统基本结构图

可见计算机控制系统完全可用图 3-32 所示的采样控制系统结构图来描述。图中，前一个采样开关即表示 A/D 采样过程，采样周期为 T；后一个采样开关即表示 D/A 转换过程，采样周期同为 T；$D(z)$ 表示控制单元脉冲传递函数；$G_h(s)$ 表示采样保持器传递函数；$G_0(s)$ 表示控制对象传递函数。

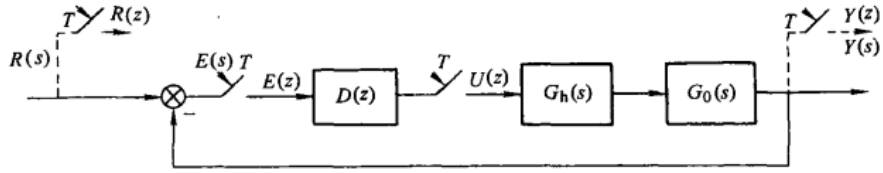


图 3-32 采样控制系统结构图

一、采样控制系统的数学描述

研究采样控制系统的数字仿真技术，同样也必须先从采样控制系统的数学描述入手。

（一）对采样信号的描述——z 变换法

设连续信号为 $x(t)$，则在复频域可用拉普拉斯变换描述为

$$X(s) = L\{x(t)\}$$

当 $x(t)$ 经采样成为脉冲信号 $x^*(t)$ 后，由下式描述

$$x^*(t) = \sum_{k=0}^{\infty} x(kT)\delta(t - kT)$$

它是一脉冲序列信号。对其进行拉普拉斯变换，得

$$X^*(s) = L\{x^*(t)\} = L\left\{\sum_{k=0}^{\infty} x(kT)\delta(t - kT)\right\} = \sum_{k=0}^{\infty} x(kT)\mathrm{e}^{-kTs}$$

可见，其为拉普拉斯算子 s 的超越函数形式，不便分析和运算，于是令 $z = e^{Ts}$，则

$$X(z) = X^*(s)\Big|_{e^{Ts}=z} = \sum_{k=0}^{\infty} x(kT)z^{-k}$$

即为 $x(t)$ 的 z 变换函数，写作

$$X(z) = Z\{x(t)\}$$

它作为关于算子 z 的多项式形式，用于描述采样信号 $x^*(t)$ 在采样时刻 $t = kT$ 的变化情况。

（二）采样时刻之间信号变化的描述——扩展 z 变换法

由以上知，普通 z 变换只描述采样时刻上信号的变化情况，但有时往往需要知道两采样

时刻之间信号的变化情况。由于普通 z 变换方法实际上已将采样时刻之间的信号信息“丢失”掉了；所以用其来描述两采样时刻之间的信号变化不再合适。因而，应采用所谓扩展 z 变换的方法。

扩展 z 变换（亦称广义 z 变换）原理已在有关课程中学习过，这里只引用其结论，简要叙述如下：

设控制对象传递函数为 $G(s)$，其脉冲响应函数为 $g(t)$，则按扩展 z 变换定义有

$$G(z,\Delta)=Z\{g(kT+\Delta T)\}=Z\{G(s)e^{\Delta Ts}\}\qquad 0<\Delta<1 \tag{3-25}$$

式中，Δ 为超前系数，表示在一个周期 T 内超前采样时刻 kT 的时间比例。$e^{\Delta Ts}$ 为虚拟超前因子。式（3-25）也被称做超前扩展 z 变换。

扩展 z 变换还可以表示为

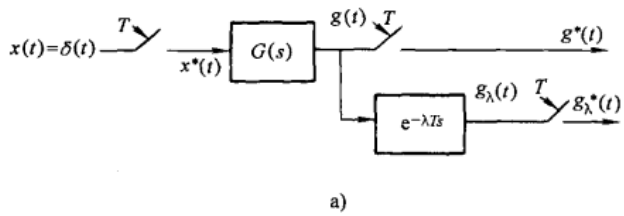
$$G(z,m)=z^{-1}Z\{G(s)e^{mTs}\}\qquad 0<m<1 \tag{3-26}$$

式中，$m=1-\lambda$，即有 $0<\lambda<1$；λ 为滞后系数，表示在一个周期内滞后采样时刻 kT 的时间比例。式（3-26）也被称做滞后扩展 z 变换。

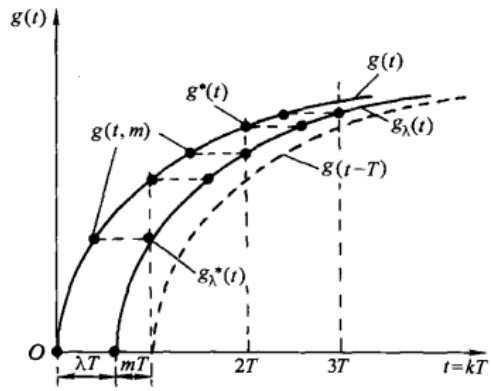
由上式继续推导可得

$$\begin{aligned}G(z,m)&=Z\{g((k+m)T-T)\}=Z\{g(kT-(1-m)T)\}\\&=Z\{g(kT-\lambda T)\}=Z\{G(s)e^{-\lambda Ts}\}\qquad 0<\lambda<1\end{aligned} \tag{3-27}$$

同样地，$e^{-\lambda Ts}$ 为虚拟滞后因子，其原理可由图 3-33 清晰、直观地表示。



a)



b)

图 3-33　扩展 z 变换示意图

扩展 z 变换的求取方法与普通 z 变换相同，典型函数的扩展 z 变换通过查表方式求取更为方便。一般情况可用极点留数法，公式为

$$G(z,m)=z^{-1}\sum_k \operatorname{res}\left[\frac{G(s)e^{mTs}}{1-e^{Ts}z^{-1}}\right]_{s=s_k}$$

式中，s_k 为 $G(s)$ 的极点，$k=1,2,\cdots,n$。

（三）对离散（采样）信号相互作用的描述——差分方程

设两离散（采样）信号分别为 $u^*(t)$、$y^*(t)$，则 $u(kT)$、$y(kT)$，或简写为 $u(k)$、$y(k)$ 为两信号的离散表达式。由控制理论知，若 $y(k)$ 是控制对象在 $u(k)$ 作用下的输出响应，则两者之间的关系可用差分方程的一般形式表述如下

$$y(k)+a_1y(k-1)+\cdots+a_ny(k-n)=b_0u(k)+b_1u(k-1)+\cdots+b_mu(k-m)\quad(n\geqslant m) \tag{3-28}$$

差分方程利用两信号在不同采样时刻的值 $y(k-j)$，$j=1,2,\cdots,n$ 和 $u(k-i)$，$i=1,2,$

…，m 之间的线性组合，表征了 $u(k)$ 对 $y(k)$ 的制约作用，不同的作用表现为不同的两组系数 a_j，b_i（$j=1,2,\cdots,n$，$i=1,2,\cdots,m$），在线性定常系统中，这些系数仅与控制对象的结构参数有关，而与输入、输出信号形式无关。

差分方程中 k 表示以周期 T 变化的离散时刻（即 $t=kT$），而 n、m 分别表示输出和输入函数的阶次，与连续系统的微分方程描述类似，阶次越高，表明系统输出受制约的因素越多，系统变化过程就越复杂，求解运算也就越困难。一般以 n 作为系统的差分阶次，并称式（3-28）为高阶差分方程。

往往为求解方便，高阶差分方程也可表示为状态方程形式。单输入-单输出情况下，状态方程一般形式为

$$\begin{cases}\boldsymbol{X}_{k+1}=\boldsymbol{A}_{\mathrm{d}}\boldsymbol{X}_k+\boldsymbol{B}_{\mathrm{d}}\boldsymbol{U}_k\\ \boldsymbol{Y}_{k+1}=\boldsymbol{C}_{\mathrm{d}}\boldsymbol{X}_{k+1}+\boldsymbol{D}_{\mathrm{d}}\boldsymbol{U}_{k+1}\end{cases}$$

式中，$\boldsymbol{X}_{k+1}=[x_{1k},x_{2k},\cdots,x_{nk}]^{\mathrm{T}}$ 为离散系统状态变量；$\boldsymbol{A}_{\mathrm{d}}$，$\boldsymbol{B}_{\mathrm{d}}$，$\boldsymbol{C}_{\mathrm{d}}$，$\boldsymbol{D}_{\mathrm{d}}$ 为离散系统状态方程各系数阵，意义与连续系统类似；其中元素的确定与系统的结构、参数有关，或可简单地说与高阶差分方程中各系数 a_j、b_i 有关。

（四）对离散（采样）信号传递关系的描述——z 传递函数

对式（3-28），两边求零初始条件下的 z 变换，可得

$$Y(z)+a_1z^{-1}Y(z)+\cdots+a_nz^{-n}Y(z)=b_0U(z)+b_1z^{-1}U(z)+\cdots+b_mz^{-m}U(z)$$

容易求得

$$G(z)=\frac{Y(z)}{U(z)}=\frac{b_0+b_1z^{-1}+\cdots+b_mz^{-m}}{1+a_1z^{-1}+\cdots+a_nz^{-n}}$$

$$=\frac{b_0z^n+b_1z^{n-1}+\cdots+b_mz^{n-m}}{z^n+a_1z^{n-1}+\cdots+a_n}\qquad(n\geqslant m)\qquad(3\text{-}29)$$

由控制理论知，式（3-29）即为两离散变量 $y(k)$、$u(k)$ 的 z 变换 $Y(z)$、$U(z)$ 的脉冲传递函数，即 z 传递函数，表示为 z 传递函数后，同样可以表示成 z 平面上的零极点增益表达式和部分分式表达式，即

$$G(z)=\frac{Y(z)}{U(z)}=K_{\mathrm{d}}\frac{\prod_{i=1}^{m}(z-z_{\mathrm{d}i})}{\prod_{j=1}^{n}(z-p_{\mathrm{d}j})}$$

$$G(z)=\sum_{j=1}^{n}\frac{r_{\mathrm{d}j}}{z-p_{\mathrm{d}j}}+h_{\mathrm{d}}(z)$$

（五）采样系统数学描述的相互转换

采样系统的几种数学描述形式，除了定义域的区别外，形式上与连续系统完全一样，相互转换的关系也与连续系统相同，因此第二章所述的连续系统数学描述的相互转换方法完全可以适用。

当采样系统分别表示为：

z 传递函数分子、分母系数向量形式

$$(\mathrm{numd},\mathrm{dend})=([b_0,b_1,\cdots,b_m],[1,a_1,\cdots,a_n])$$

零极点增益向量形式

$$(\mathrm{Zd},\mathrm{Pd},\mathrm{Kd})=([z_{d1},\cdots,z_{dm}],[p_{d1},\cdots,p_{dn}],K_d)$$

部分分式向量形式

$$(\mathrm{Rd},\mathrm{Pd},\mathrm{hd})=([r_{d1},\cdots,r_{dn}],[p_{d1},\cdots,p_{dn}],h_d)$$

还有离散状态方程各系数矩阵形式

$$(\mathrm{Ad},\mathrm{Bd},\mathrm{Cd},\mathrm{Dd})$$

几种形式之间均可利用 MATLAB 语言控制系统工具箱中的数学模型转换函数 tf2ss（　）、tf2zp（　）、ss2tf（　）、ss2zp（　）、zp2tf（　）、zp2ss（　）以及 residue（　）等作相互转换，因离散系统模型转换过程中遵循的原则和采用的算法与连续系统没有区别，仅在各参量的表达意义上有所不同，故与在连续系统下应用此类函数完全相同。

在 MATLAB5. x 以上版本中，运用 LTI 对象数据类型概念，建立离散（采样）系统模型更为方便，只要将已知的系统参数向量按照以下调用格式输入，即可获得离散（采样）系统的 LTI 对象各种形式的模型：

sys = tf（*numd*，*dend*，Ts）

sys = ss（*Ad*，*Bd*，*Cd*，*Dd*，Ts）

sys = zpk（*Zd*，*Pd*，*Kd*，Ts）

调用时应特别注意输入正确的采样周期 Ts，否则将对模型精度影响极大。一旦 LTI 对象数据类型模型建立起来，可采用与连续系统下相同的各种模型转换函数进行不同模型之间的相互变换，以适应不同仿真、分析方法的需求。此方面深入的学习请参阅书后所附参考文献。

二、采样控制系统的仿真方法

（一）差分方程递推求解法

对图 3-34，已知 $D(z)=\dfrac{U(z)}{E(z)}$，则 $U(z)=D(z)E(z)$。当 $D(z)$形如

$$D(z)=\frac{d_0+d_1z^{-1}+\cdots+d_rz^{-r}}{1+c_1z^{-1}+\cdots+c_lz^{-l}}$$

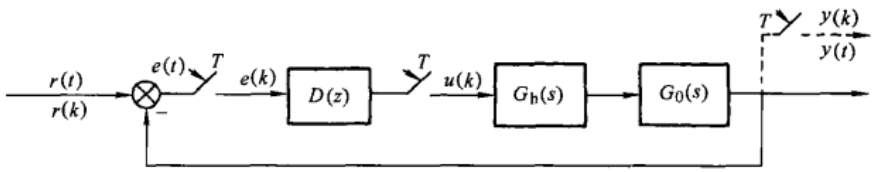


图 3-34　采样控制系统典型结构图

时，很方便得到

$$(1+c_1z^{-1}+\cdots+c_lz^{-l})U(z)=(d_0+d_1z^{-1}+\cdots+d_rz^{-r})E(z)$$

求 z 反变换，并整理得到递推式

$$u_k=-c_1u_{k-1}-\cdots-c_lu_{k-l}+d_0e_k+d_1e_{k-1}+\cdots+d_re_{k-r}$$

相当于一个多步法递推算式，只要 u_k 和 e_k 的前若干步值已知，就可以递推得到 u_k。

同样，已知 $$G_0(s)=\frac{Y(s)}{U(s)}$$

容易求出 $$G(z)=\frac{Y(z)}{U(z)}=Z\{G_h(s)G_0(s)\}=\frac{b_0+b_1z^{-1}+\cdots+b_mz^{-m}}{1+a_1z^{-1}+\cdots+a_nz^{-n}}$$

也能得到

$$y_k=-a_1y_{k-1}-\cdots-a_ny_{k-n}+b_0u_k+b_1u_{k-1}+\cdots+b_mu_{k-m}$$

再考虑

$$e_k=r_k-y_k$$

就能按信号传递过程，从参考输入 r_k 开始，逐步求得各部分解 e_k、u_k 和输出 y_k。

仿真程序框图如图 3-35 所示。图中运行参数 T 为采样周期（仿真步长）；r 为输入幅值；T_N 为终止时间。

差分方程描述的就是离散各量采样时刻点上的相互关系和变化情况，因此当仿真步长取采样系统的实际采样周期为 T 时，求取的结果无截断误差，从理论上说算法是精确的。

这种方法简便易行，只要 $D(z)$、$G(z)$ 已知，则仿真过程非常简单，且无截断误差，结果可靠。缺点是当系统复杂时，$G(z)$ 难以求取，即使求出 $G(z)$ 也无法观察系统中其他中间变量的响应特性，也不便考虑有非线性影响的情况。

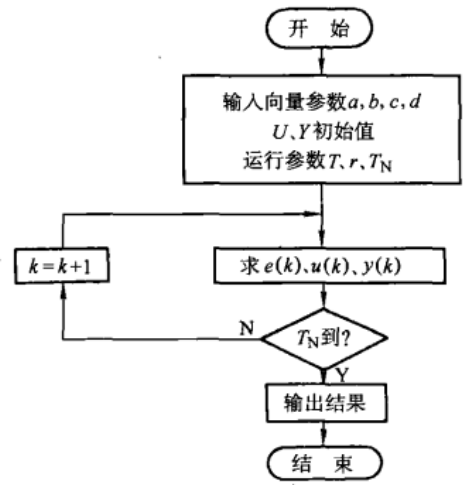


图 3-35 差分方程递推求解仿真程序框图

（二）连续部分按环节离散化

当系统连续部分较复杂时，不必去化简和求取 $G(z)$，而按照连续系统环节离散化仿真方法，将连续部分中各环节经离散化处理后，与采样部分一并考虑进行仿真。

图 3-36 中连续部分各环节之间虚设采样开关和保持器，按环节离散化方法建立模型，应取数量级小于采样周期 T 较多的仿真步长 h，才能很好地反映出连续系统在离散信号作用下，各环节在 T 内的细微变化。取得恰当时，仿真结果精度很好，更能反映这类系统各项性能的真实变化情况。

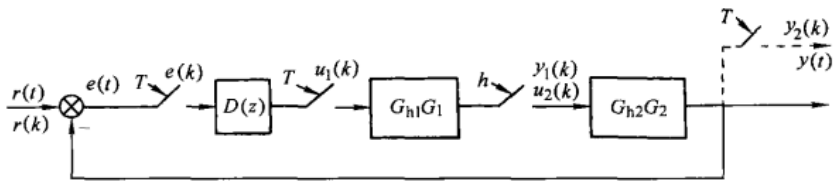


图 3-36 连续部分按环节离散化典型结构图

一般取 $T\gg h$，且 T 为 h 的整倍数最便于处理。在每个采样周期 kT 时刻，离散环节信号的作用经实际采样开关传递到连续部分，并保持一个周期。在这段周期内以步长 $t=h$ 计算连续部分各环节的变化情况，直到下一采样周期 $t=(k+1)T$ 时刻，发生变化了的新的离

散环节信号再次传递给连续部分，如此循环，直至 $t=T_N$ 仿真过程结束。其仿真框图如图 3-37 所示。

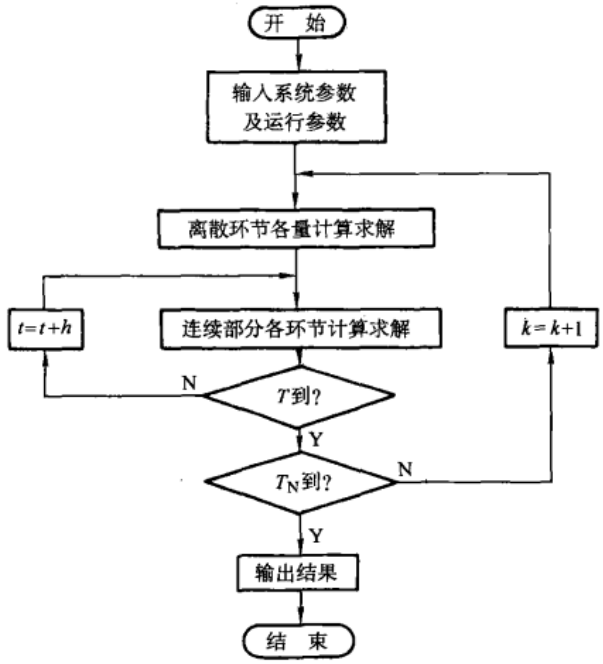


图 3-37　连续部分采用环节离散化方法的仿真框图

（三）控制器设计为连续系统环节 $D(s)$

采样系统的控制器有时设计为连续系统环节形式，其传递函数为 $D(s)$（如各类有源、无源校正网络等），但其发生的作用又是每相隔一个采样周期 T 才经采样开关传递给控制对象连续部分。在 T 时刻内，$D(s)$、$G_0(s)$ 同时都在按照自己的规律连续变化，如图 3-38 所示。

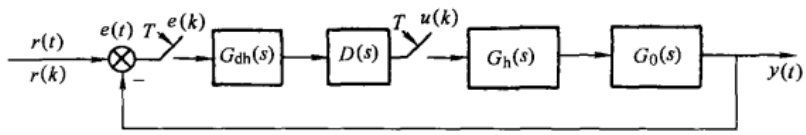


图 3-38　控制器为 $D(s)$ 形式的典型结构图

可见，对这类系统仿真，要顾及到两部分连续系统各环节 $D(s)$、$G_0(s)$ 的实际变化过程，对两部分都应正确地通过仿真得到准确的结果，并在每个采样周期 T 到来时将变化结果及时传递到相应环节去，这样所得采样控制系统仿真结果才合理且有意义，仿真过程如图 3-39 所示。

由图 3-39 可知，在采样时刻 kT 与 $(k+1)T$ 之间，连续部分一直在变化，故必须以步长 $t=h$（选取原则同上小节）同时仿真计算所有连续部分各环节 $D(s)$、$G_0(s)$ 的输出。当采样时刻 $t=(k+1)T$ 到来时，将信号变化结果经系统的采样开关传递到相应环节，这时所有环节才被全部连接起来，进行整个系统离散（采样）部分的数值计算，因此能得到整个采样控制系统较准确的仿真数据结果。

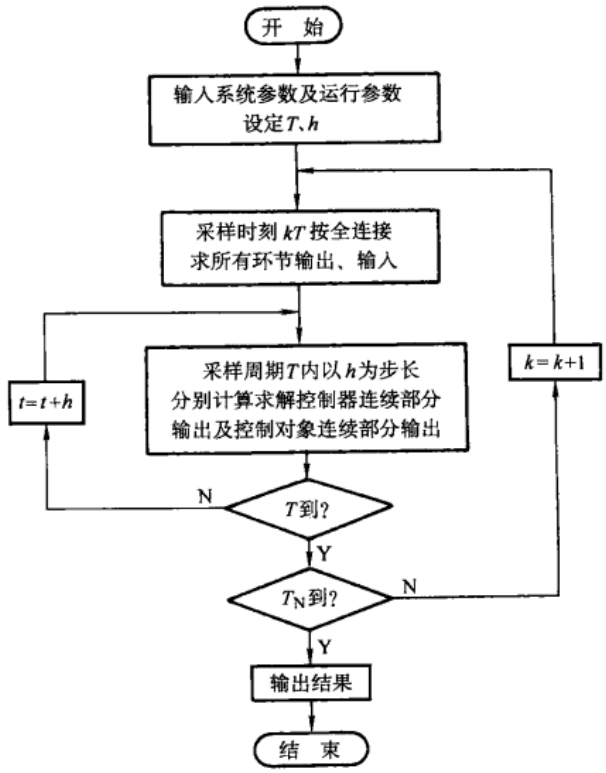


图 3-39 控制器为 $D(s)$ 形式的系统仿真框图

三、采样控制系统仿真程序实现

基于以上分析，采样系统的仿真程序不难编制，采用 MATLAB 语言更易于实现。考虑到采样系统有与连续系统不同的特点，在编制程序中应针对一些特殊情况考虑程序的实现问题。

（一）高阶差分方程的仿真程序

若采样系统直接给出输入-输出闭环 z 传递函数 $G_B(z)$ 形式，即

$$G_B(z)=\frac{Y(z)}{U(z)}=\frac{b_0+b_1z^{-1}+\cdots+b_mz^{-m}}{1+a_1z^{-1}+\cdots+a_nz^{-n}}$$

一般总有 $n\geqslant m$，则相应差分方程

$$y(k)+a_1y(k-1)+\cdots+a_ny(k-n)=b_0u(k)+b_1u(k-1)+\cdots+b_mu(k-m)$$

为高阶差分方程。解该方程，立即可得系统的采样时刻输出响应 $y(k)$。

若分别得到控制器 z 传递函数 $D(z)$ 和控制对象 z 传递函数 $G(z)$，如前文第二小节中所述形式，则需求解一组高阶差分方程

$$\begin{cases}e_k=r_k-y_k\\ u_k=-c_1u_{k-1}-\cdots-c_lu_{k-l}+d_0e_k+d_1e_{k-1}+\cdots+d_re_{k-r}\\ y_k=-a_1y_{k-1}-\cdots-a_ny_{k-n}+b_0u_k+b_1u_{k-1}+\cdots+b_mu_{k-m}\end{cases}$$

可见，以上算法最终都归结为求解高阶差分方程问题。在计算机上实现高阶差分方程的求解程序，应当注意以下问题：

1）建立向量存储单元，保存和记忆输入 u_k、输出 y_k 前若干时刻的值 u_{k-1}，…，u_{k-m}和 y_{k-1}，…，y_{k-n}。

2）每运算一个时刻值后，要及时刷新和摒弃相应的存储单元内容。即只保留 u_k 及其前 m 个时刻的数值，y_k 及其前 n 个时刻的数值，因此，要安排相应的平移操作程序。

高阶差分方程的求解过程可用图 3-40 示意。

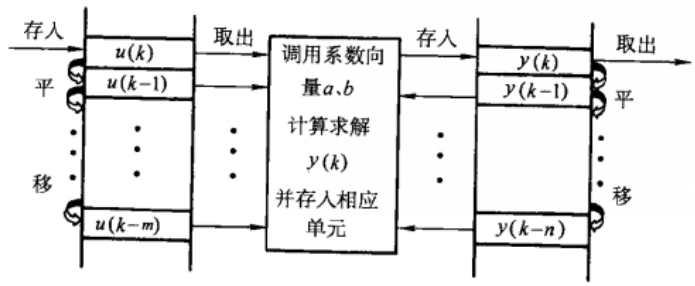


图 3-40　高阶差分方程的仿真求解过程示意图

可见，高阶差分方程求解的操作步骤是“存入→平移→取出→计算”，在求解运算主程序中，插入以下简单 MATLAB 语句即可方便地实现：

```
U = [uk;U(1:m)];          % 存入→平移
yk = -a'*Y+b'*U;          % 取出→计算
Y = [yk;Y(1:n-1)];        % 存入→平移
yt = [yt,yk];             % 输出响应数据向量
```

运行这几条语句前，注意输入 uk、存储向量 U、Y 及差分方程系数向量 a、b 等单元在主程序中均应已赋值和初始化。对控制器高阶差分方程的求解与此完全相同，只是必须注意在主程序中一定要建立正确的变量单元及存储向量单元的维数。

（二）纯滞后环节的处理方法及仿真程序

采样控制系统中常见的一些控制对象包含有滞后（延迟）环节。其数学模型为

$$D(s)=\frac{Y(s)}{U(s)}=\mathrm{e}^{-\tau s}$$

即

$$y(t)=u(t-\tau)$$

其意义为，信号 $y(t)$ 与信号 $u(t)$ 变化形式完全一样，只是滞后了一段时间 τ，如图 3-41 所示。将 $y(t)$ 离散化为 $y(kT)$，并将滞后时间常数 τ 表示为 T 的函数，则

$$\tau=(M_1+M_2)T$$

式中，M_1 为整数；M_2 为（0，1）之间的小数。

于是

$$y(kT)=u[kT-(M_1+M_2)T]$$

简单地有

$$y_k=u_{k-(M_1+M_2)}$$

以下分两种情况讨论：

1）当 $M_2=0$ 时，滞后环节时间常数 τ 为采样周期 T 的整倍数，信号 y_k 滞后 u_k 整 M_1 个

节拍，这在仿真程序中很容易实现。环节输出 y_k 的当前值，实际上恰为环节输入 u_k 的前 M_1 步值 u_{k-M_1}，因此，在计算中将 u_k 逐次保存下来，记忆前 M_1 步值，共需要 M_1+1 个存储单元，求 y_k 时，只要取出所记忆的前 M_1 步值 u_{k-M_1} 送入 y_k 单元即可，如图 3-42 所示。

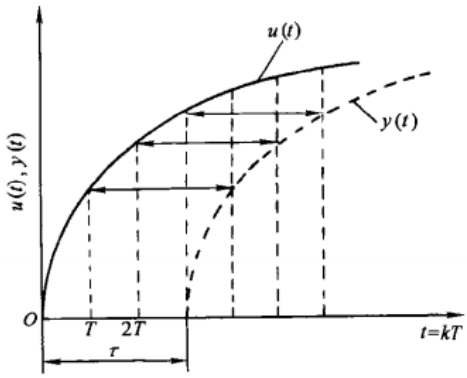


图 3-41 纯滞后环节的响应特性

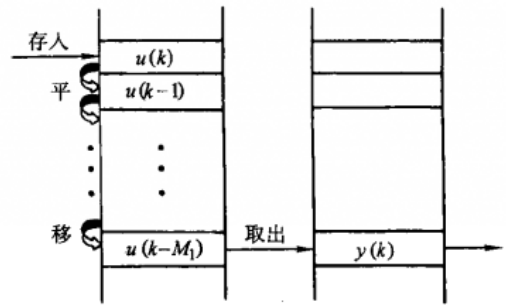


图 3-42 τ 为采样周期 T 整倍数的仿真过程示意图

为进一步运算作准备，还应在 u_{k-M_1} 取出后，将所有被存储 u 值都平移一个单元，以备再用。故仿真编程只需遵从“存入→平移→取出”的基本步骤即可。用 MTALAB 语言实现此步骤很简单，程序如下：

```
ylag = zreos(M1 + 1,1);          % 建立 M1 + 1 维存储向量,且初始值均为零
ylag = [uk;ylag(1:M1)];          % 存入当前 uk 值,平移保留前 M1 步 u 值
yk = ylag(M1 + 1);               % 取出 u(k-M1) 作为 yk
```

运行这一小段程序前，M_1、u_k 需事先给定。

2）当 $M_2 \neq 0$ 时，滞后环节时间常数 τ 不为采样周期 T 的整倍数，仿真实现要复杂些。从表达式

$$y(k) = u[k-(M_1+M_2)]$$

看出，$y(k)$ 应在 $u(k)$ 的前 M_1 步值 $u(k-M_1)$ 和前 M_1+1 步值 $u[k-(M_1+1)]$ 之间。但离散系统用差分方程表示，可供参考使用的数据只有所记忆的前 M_1+1 步离散时刻各点上的 u 值。因此，求取 $y(k)$，需通过对 $u(k-M_1)$ 和 $u[k-(M_1+1)]$ 两值的插补运算求得，如图 3-43 所示。

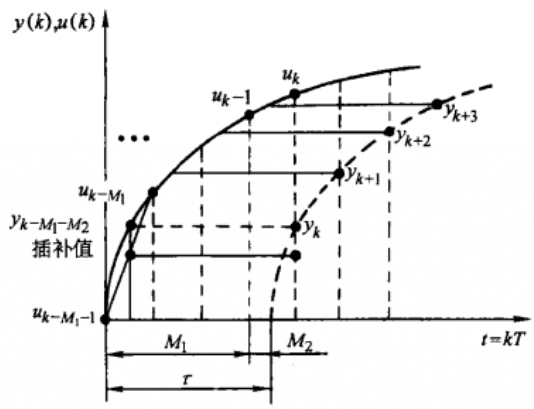


图 3-43 τ 不为采样周期 T 整倍数的直线插补方法

直接采用直线插补公式为

$$y(k) = \frac{u(k-M_1) - u(k-(M_1+1))}{T}(1-M_2)T + u(k-(M_1+1))$$

整理，得

$$y(k) = (1-M_2)u(k-M_1) + M_2 u(k-(M_1+1))$$

或更简单地写为

$$y_k = (1-M_2)u_{k-M_1} + M_2 u_{k-(M_1+1)}$$

仿真实现也不难，其基本步骤仍如“存入→平移→取出→计算”形式，仅多了一个插补运算过程，如图 3-44 所示。

下面是其 MATLAB 语言程序。

```
ylag = zeros(M1 + 2,1);                    % 建立 M1 + 2 维存储向量
ylag = [uk;ylag(1:M1 + 1)];                % 存入→平移,保留前 M1 + 1 步 u 值
yk = (1 - M2) * ylag(M1 + 1) + M2 * ylag(M1 + 2);   % 插补运算后,取出输出结果
```

其实前述两种情况可以共用上面的程序，即当 $M_2 = 0$ 时，也完全适用。同样，在进入该程序前，M_1、M_2 及 u_k 需事先给定。实际程序中，是采用自定义一段功能函数程序来实现的，需要时在主程序中调用即可。当纯滞后环节与其他典型线性环节串联时，可应用上述方法方便地求解。

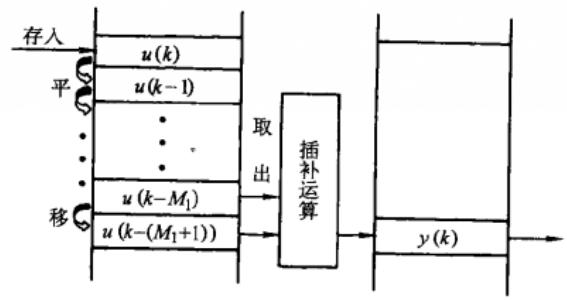


图 3-44　直线插补的仿真过程示意图

例 3-3　采样系统开环结构如图 3-45 所示。已知

$$G_h(s) = \frac{1 - e^{-Ts}}{s}$$　　零阶保持器

$$G_0(s) = \frac{\frac{K_0}{T_0}}{s + \frac{1}{T_0}}$$　　一阶惯性环节

并注意 $e^{-\beta Ts}$，$\beta = M_1 + M_2$，其定义如前述。求输出响应 $y(k)$。

解　先求 $\bar{y}$ 与输入 u 之间的 z 传递函数

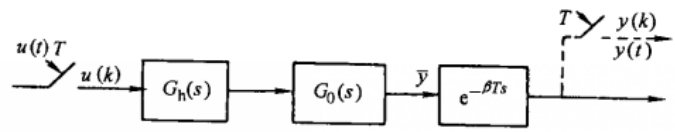


图 3-45　例 3-3 采样系统开环结构图

$$G_1(z) = \frac{\bar{Y}(z)}{U(z)} = Z\left\{\frac{1 - e^{-Ts}}{s}\cdot\frac{\frac{K_0}{T_0}}{s + \frac{1}{T_0}}\right\} = (1 - z^{-1})Z\left\{\frac{\frac{K_0}{T_0}}{s\left(s + \frac{1}{T_0}\right)}\right\}$$

$$= K_0\frac{(1 - e^{-\frac{T}{T_0}})z^{-1}}{1 - e^{-\frac{T}{T_0}}z^{-1}} \qquad (3\text{-}30)$$

再看输出 y 与 $\bar{y}$ 之间的 z 传递函数

$$G_2(z)=\frac{Y(z)}{\overline{Y}(z)}=Z\{e^{-\beta Ts}\}=z^{-\beta} \tag{3-31}$$

由式(3-30)得 $$\overline{y}(k)=e^{-\frac{T}{T_0}}\overline{y}(k-1)+K_0(1-e^{-\frac{T}{T_0}})u(k-1)$$

由式(3-31)得 $$y(k)=\overline{y}(k-\beta)=\overline{y}(k-(M_1+M_2))$$

按以上递推关系式,先求出 $u(k-1)$ 作用下的 $\overline{y}(k)$,再根据 M_1、M_2 滞后节拍即可求得 $y(k)$。

(三) 扩展 z 变换的处理方法及仿真程序

上例中,当 $M_2\neq 0$ 时,实际上是在求 $y(k)$ 的采样时刻点之间的值,虽然用直线插补方法较简单,但精度相对差一些。前节所述的扩展 z 变换方法在此也可用来解决滞后环节时间常数 β 不为采样周期 T 整倍数的仿真问题。由式(3-30)、式(3-31)知

$$G(z)=\frac{Y(z)}{U(z)}=K_0\frac{(1-e^{-\frac{T}{T_0}})z^{-1}}{1-e^{-\frac{T}{T_0}}z^{-1}}z^{-\beta} \tag{3-32}$$

先看 $M_2=0$ 时的情况。此时 $\beta=M_1$,显然可得

$$y(k)=e^{-\frac{T}{T_0}}y(k-1)+K_0(1-e^{-\frac{T}{T_0}})u(k-M_1-1) \tag{3-33}$$

而 $M_2\neq 0$ 时,$\beta=M_1+M_2$,且 $0<M_2<1$,则式(3-32)可写成为

$$G(z)=\frac{Y(z)}{U(z)}=K_0\frac{(1-e^{-\frac{T}{T_0}})z^{-(1+M_1)}}{1-e^{-\frac{T}{T_0}}z^{-1}}z^{-M_2}$$

相当于扩展 z 变换中,滞后因子 $\lambda=M_2$ 的情况,令 $m=1-\lambda=1-M_2$,则利用扩展 z 变换式(3-26)、式(3-27),有

$$G(z,m)=z^{-1}Z\{G(s)e^{mTs}\}$$

其中 $$G(s)=\frac{\frac{K_0}{T_0}(1-e^{-Ts})e^{-M_1Ts}}{s\left(s+\frac{1}{T_0}\right)}$$

采用留数法 $$G(z,m)=z^{-1}\sum_k \operatorname{res}\left[\frac{\frac{K_0}{T_0}(1-e^{-Ts})e^{-M_1Ts}}{s\left(s+\frac{1}{T_0}\right)}\cdot\frac{e^{mTs}}{1-e^{Ts}z^{-1}}\right]_{z=s_k}$$

在本例中,$k=2$,即得 $s_1=0$,$s_2=-1/T_0$,代入上式,求出

$$G(z,m)=z^{-1}K_0(1-z^{-1})z^{-M_1}\left(\frac{1}{1-z^{-1}}-\frac{e^{-m\frac{T}{T_0}}}{1-e^{-\frac{T}{T_0}}z^{-1}}\right)$$

$$=\frac{K_0z^{-(1+M_1)}[(1-e^{-m\frac{T}{T_0}})-(e^{-\frac{T}{T_0}}-e^{-m\frac{T}{T_0}})z^{-1}]}{1-e^{-\frac{T}{T_0}}z^{-1}}$$

所以,据上式,由 $$Y(z,m)=G(z,m)U(z)$$

得　$(1-e^{-\frac{T}{T_0}}z^{-1})Y(z,m)=K_0z^{-(1+M_1)}[(1-e^{-m\frac{T}{T_0}})-(e^{-\frac{T}{T_0}}-e^{-m\frac{T}{T_0}})z^{-1}]U(z)$

故易得输出递推方程为

$$y(k,m)=e^{-\frac{T}{T_0}}y(k-1,m)+K_0(1-e^{-m\frac{T}{T_0}})u(k-M_1-1)+K_0(e^{-m\frac{T}{T_0}}-e^{-\frac{T}{T_0}})u(k-M_1-2) \tag{3-34}$$

当 m 在［0，1］之间取值时，求解上式即得采样时刻 $(k-1)T$ 到 kT 之间的输出响应值。特殊地，$m=1$，即 $M_2=0$ 时，为滞后整倍数 M_1 的情况，该式与普通 z 变换求得的递推公式完全相同。

实际仿真中，只要滞后时间常数 β 已知，则其整倍数部分 M_1 和小数部分 M_2 就被确定，同时 $m=1-M_2$ 也被确定，代入式（3-34），就能求得 $y(k, m)$ 值。因下一时刻 $t=(k+1)T$ 还需求采样时刻之间的输出响应值 $y(k+1, m)$，递推公式中还要用到 $y(k, m)$ 值，故应建立专门的存储单元将 $y(k, m)$ 记忆下来。当要观察采样时刻之间多个点上的响应值时，则应将各点确定的 m 值存入相应的向量单元，然后每一步计算都将各点值存储记忆，以便下一步计算时使用这些数据。

对本例，需建立的存储单元有

U = ［u_k, u_{k-1}, …, u_{k-M_1-1}, u_{k-M_1-2}］′; M_1+3 维列向量

Ykb = ［y_{k-1}, $y_{k-1,m}$］′; 维数与 m 有关的列向量，存储 y_k 的前一次值 y_{k-1} 和采样时刻之间的前一次值 $y_{k-1,m}$

Yk = ［y_k, $y_{k,m}$］′; 维数与 m 有关的列向量，存储 y_k 的当前值和采样时刻之间的当前值 $y_{k,m}$

与 m 有关的各系数可离线计算存入相应单元，也可在线计算。视对系统计算速度的要求和存储单元容量而定。本例中，采用在线方式根据需要计算求取。

考虑当前输出值 y_k 和滞后 βT 时的输出值 $y_{k,m}$，将式（3-33）、式（3-34）合写为

$$\begin{bmatrix}y_k\\y_{k,m}\end{bmatrix}=e^{-\frac{T}{T_0}}\begin{bmatrix}y_{k-1}\\y_{k-1,m}\end{bmatrix}+\begin{bmatrix}K_0(1-a) & 0\\K_0(1-a_m) & K_0(a_m-a)\end{bmatrix}\begin{bmatrix}u_{k-M_1-1}\\u_{k-M_1-2}\end{bmatrix}$$

即　$\boldsymbol{Y}_k=a\boldsymbol{Y}_{kb}+\boldsymbol{b}\boldsymbol{U}_{km}$

其中　$a=e^{-\frac{T}{T_0}}$; $a_m=e^{-m\frac{T}{T_0}}$;

$$\boldsymbol{b}=\begin{bmatrix}K_0(1-a) & 0\\K_0(1-a_m) & K_0(a_m-a)\end{bmatrix};$$

$$\boldsymbol{u}_{km}=\begin{bmatrix}u_{k-M_1-1}\\u_{k-M_1-2}\end{bmatrix}$$

仿真程序框图如图 3-46 所示。

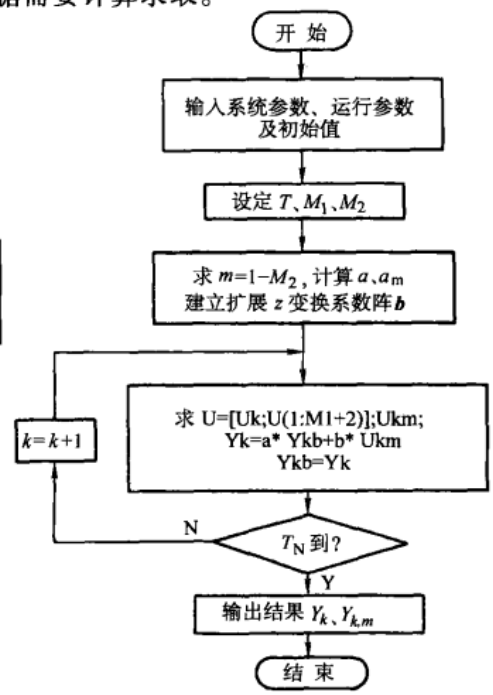


图 3-46　扩展 z 变换仿真程序框图

用 MATLAB 语言编程如下：

```
m = 1 - M2;                                  % 求超前系数 m = 1 - M2
Yk = zeros(2,1); Ykb = Yk;                   % 存储单元初始化
    U = zeros(M1 + 3,1);
    a = exp( - T/T0); am = exp( - m * T/T0);  % 计算扩展 z 变换系数
b = [K0 * (1 - a),0;K0 * (1 - am),K0 * (am - a)]; % 形成运算系数阵,阵中首行对应普通 z 变换求解
                                             % 下面各行分别对应扩展 z 变换求解
ykm = zeros(1,2); Tk = 0;                    % 结果存储单元初始化
    for k = 1:N
      U = [uk;U(1:M1 + 2)];                  % 存入当前 uk 值并平移
      Ukm = U(M1 + 2:M1 + 3);                % 取 uk 前 M1 - 1 和 M1 - 2 次值
        Yk = a * Ykb + b * Ukm;              % 取各时刻、各点 y、u 值运算求解
        Ykb = Yk;                            % 记录前次值
        ykm = [ykm;Yk'];                     % 存储各时刻、各点结果数据
        Tk = [Tk,k * T];                     % 建立采样时刻时间序列值向量
    end                                      % 计算时间到则停止运行并转输出
plot(Tk,ykm), hold on                        % 输出 yk 和 yk,m 结果曲线
```

以上程序运行前，需先知系统参数 K_0、T_0、M_1、M_2 及运行参数 T 和 u_k 值。

第五节　问题与探究——一类非线性控制系统数字仿真的效率问题

一、问题提出[88]

起重机作为一种搬运工具，在工业生产中发挥着重要的作用。但是，由于起重机自身结构的原因，使得货物在吊运过程中不可避免会产生摆动；如何有效地消除货物在吊运过程中的摆动以提高起重机的工作效率是长期以来国内外控制领域研究的一个典型问题[89]。变结构控制理论是一种非线性控制理论，自 20 世纪 50 年代于莫斯科诞生以来，经过了半个多世纪的完善和发展。特别是近年来，滑模变结构控制理论得到了国内外控制界普遍关注和重视。滑模变结构控制器控制规律简单，对系统的数学模型精确性要求不高，可以有效地调和动、静态之间的矛盾且鲁棒性较强，近年来已被广泛应用于处理一些复杂的非线性、时变、多变量耦合及不确定系统[90]。

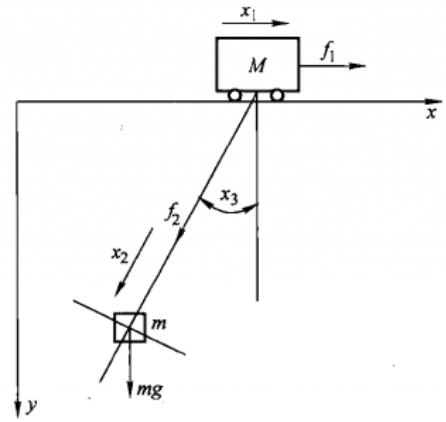


图 3-47　起重机的物理模型

实际的起重机系统比较复杂，除了元件的非线性外，还受到各种干扰，如小车与导轨之间的摩擦、风力的影响等。为了便于分析，对实际系统进行简化，图 3-47 所示为典型的起重机系统物理模型。针

对绳长固定情况，这里讨论时采用二维滑模变结构控制器，同时对起重机的水平位置和摆绳角度进行控制。图 3-48 所示为控制系统的 Simulink 仿真模型[91]。

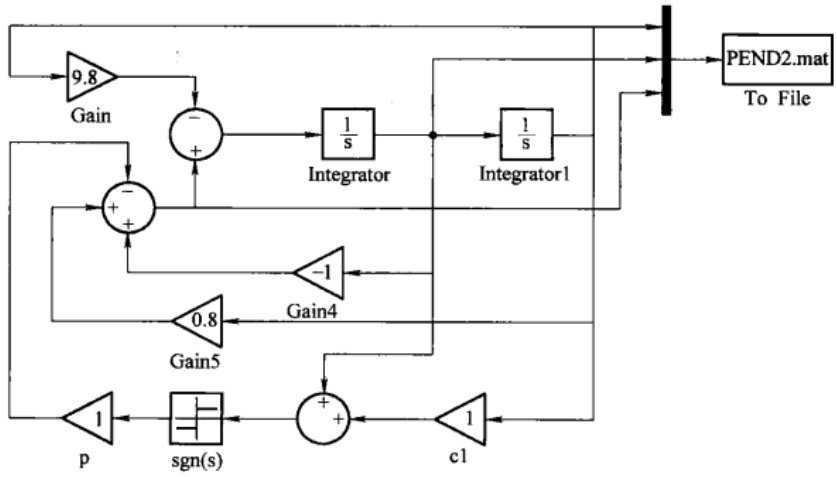


图 3-48　二维滑模变结构控制器仿真模型

采用 Variable-step 的 ODE45 算法进行 Simulink 仿真，结果如图 3-49 所示。可见，当系统仿真时间在 1s 左右时，速度非常慢，仿真过程停滞不前。

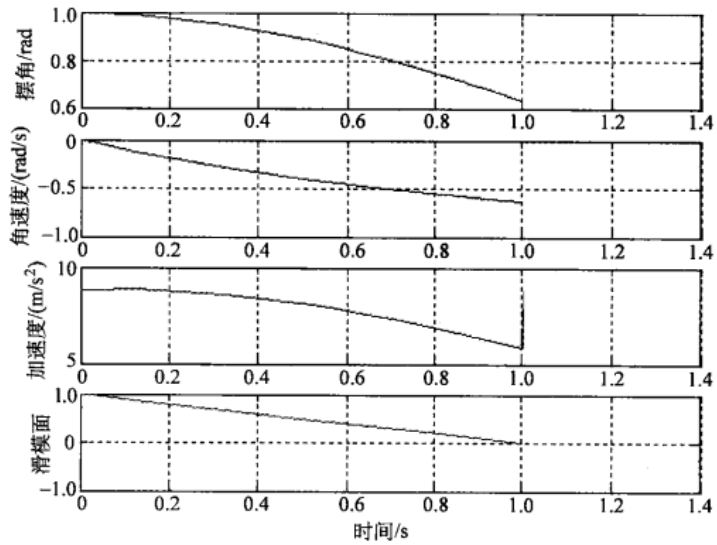


图 3-49　仿真结果

本节将针对上述现象与问题，讨论一类非线性控制系统数字仿真的效率问题。

二、问题分析

（一）过零检测[78]

MATLAB/Simulink 在仿真过程中存在“过零检测”问题。过零检测是通过在系统和求解器之间建立对话的方式工作，对话包含的一个内容是事件通知。事件由过零表示，过零在下列两个条件下产生：

1）信号在上一个时间步改变了符号（含变为 0 和离开 0）。

2）模块在上一个时间步改变了模式。

过零是一个重要的事件，表征系统中的不连续性。如果仿真中不对过零进行检测，可能会导致不准确的仿真结果。当采用变步长求解器时，Simulink 能够检测到过零（使用固定步长的求解器，Simulink 不检测过零）。当一个模块通知系统前一时间步发生了过零，变步长求解器就会缩小步长，即便绝对误差和相对误差是可接受的。缩小步长的目的是判定事件发生的准确时间。当然，这样会降低仿真的速度，但这样做对有些模块来讲是至关重要和必要的，因为这些模块的输出可能表示了一个物理值，它的零值有着重要的意义。事实上，只有少量的 Simulink 模块能够发出过零事件通知，如图 3-50 所示。

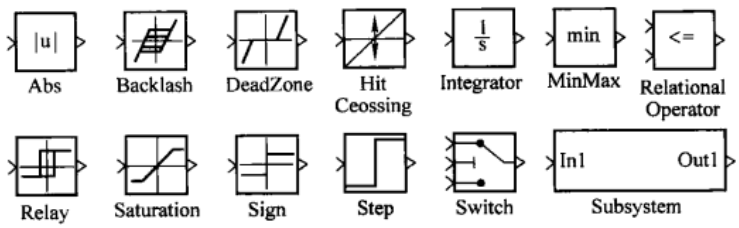


图 3-50 能够产生过零通知的 Simulink 模块

（二）系统仿真停滞的原因

由图 3-48 可知，系统仿真模型中存在不连续模块 Signum，当系统于 1s 左右到达滑模面（$s=0$）时，Signum 模块向系统发出过零通知。而当采用变步长求解器时，Simulink 能够检测到过零状态。如图 3-51、图 3-52 所示，由于滑模面在 1s 处不能正常归零，所以 Signum 模块就反复过零，同时一直向求解器发出过零通知。求解器便相应地一直不停地缩小步长，如图 3-53 所示，系统大约经过 12 个仿真步到达 1s 处时，步长急剧缩小至接近于零。这样，由于仿真步长太小，系统便在不连续处形成了过多的点，超出了系统可用的内存和资源，使得系统进展缓慢，仿真停滞不前[91]。

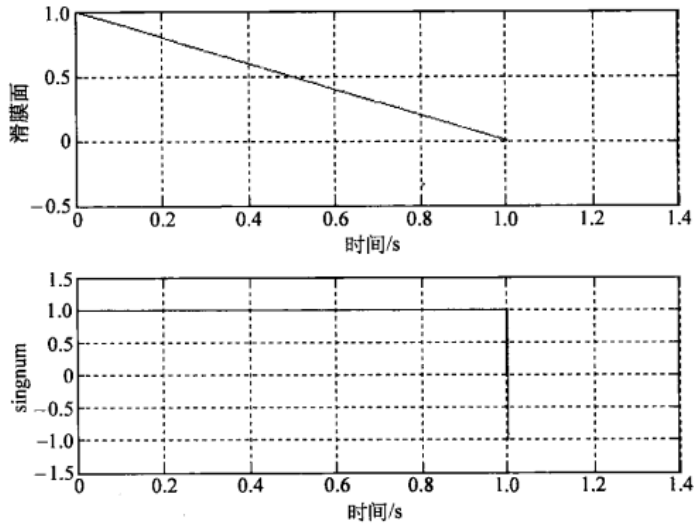


图 3-51 滑模面和 Signum 模块的时域响应

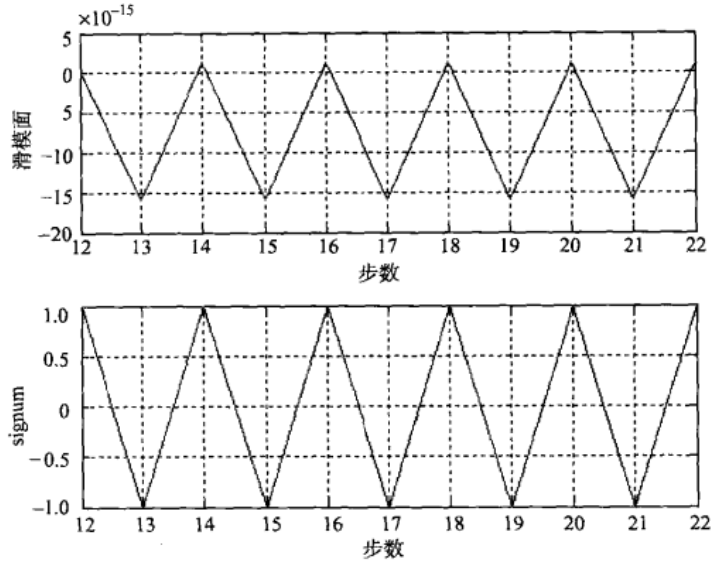


图 3-52　1s 左右时的滑模面和 Signum 响应的局部放大图

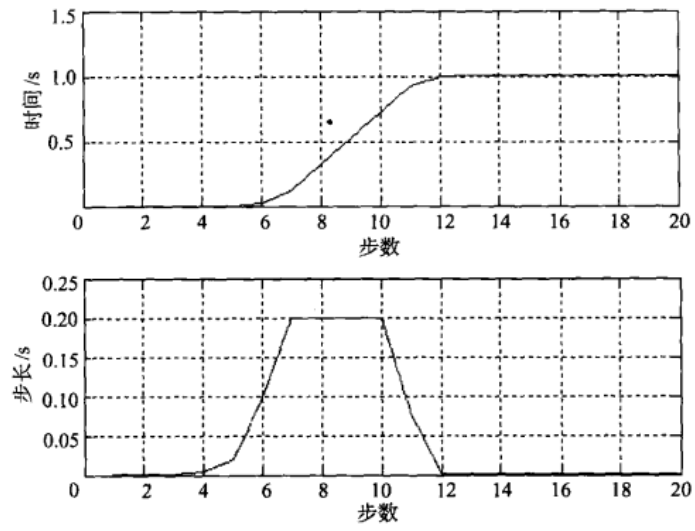


图 3-53　系统仿真步长

（三）提高仿真效率的方法

（1）速度慢的原因　通过以上分析，我们总结、概括得出了以下几种系统仿真速度慢的原因：

1）系统方程中存在不连续函数 sign(s)。

2）Simulink 仿真模型中存在能够产生过零通知的 signum 模块。

3）采用的变步长求解器具有过零检测并自动调整步长的功能。

（2）解决策略　基于以上原因，我们提出了以下四种解决策略[88]：

1）采用不能够产生过零通知的 Fcn 函数模块。

2）取消 Zero crossing detection 功能。

3）采用 fixed-step 求解器。

4）柔化 sgn(s) 函数，使其连续化。

这四种解决策略，单独任何一种或几种都可行。图 3-54 所示为采用策略 1）的系统 Simulink 仿真模型，图 3-55、图 3-56 所示为仿真结果。

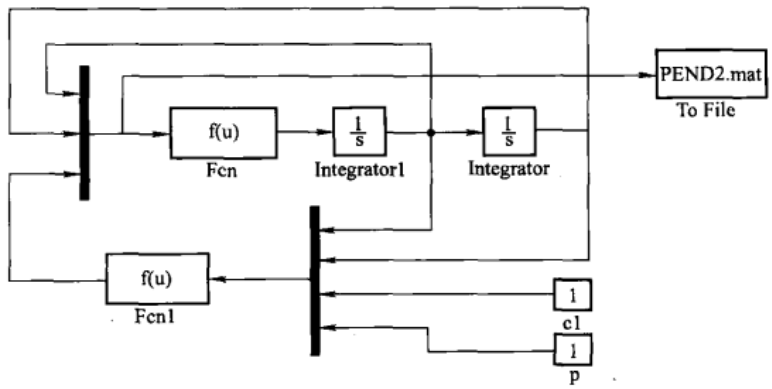


图 3-54 采用 Fcn 函数模块后的系统仿真模型

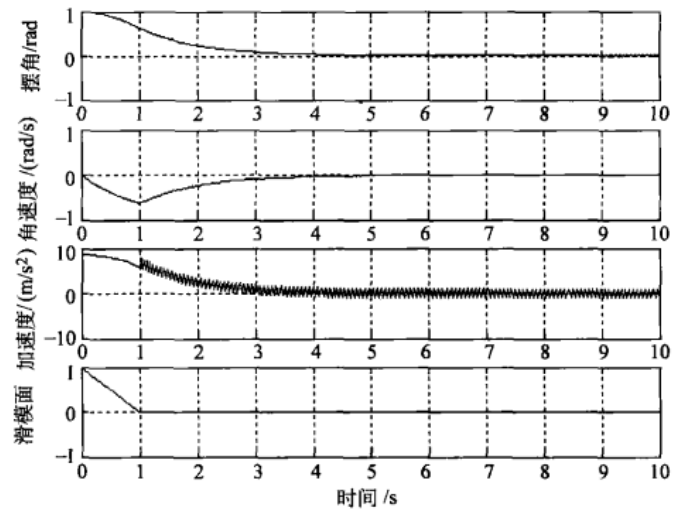


图 3-55 系统时域图

由于 Fcn 模块不支持过零，所以系统在不连续的情况下仍然能迅速完成仿真。除此之外，我们还采用了 fixed-step 求解器和变步长下置 Zero crossing detection 为 off 的仿真方法，其仿真结果也如图 3-55、图 3-56 所示。可见，前三种方法的仿真结果一样。

由上面的仿真结果可以看出：取消系统的过零检测功能之后，仿真速度快；但是由于仍然存在不连续模块 signum，所以加速度存在较大的抖振。抖振现象使得控制系统难以工程实现，为了能消除系统因为不连续性而存在的抖振问题，同时加快仿真速度，且不影响系统的仿真效果，我们设计了第四种控制策略——柔化不连续的 sgn(s) 函数，使其连续化。原 sgn(s) 函数如图 3-57a 所示，重构后的 sat(s) 函数如图 3-57b 所示。

其中，sat(s) 函数的表达式为

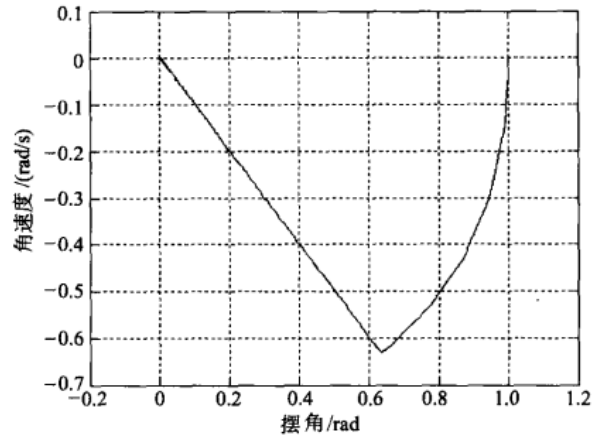


图 3-56　系统相平面图

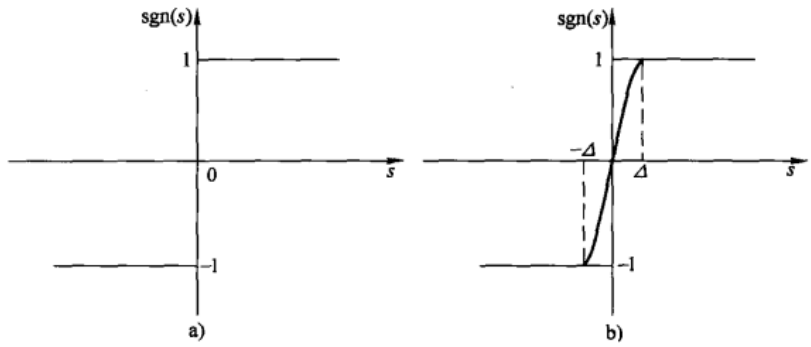


图 3-57　sgn(s) 和 sat(s) 函数

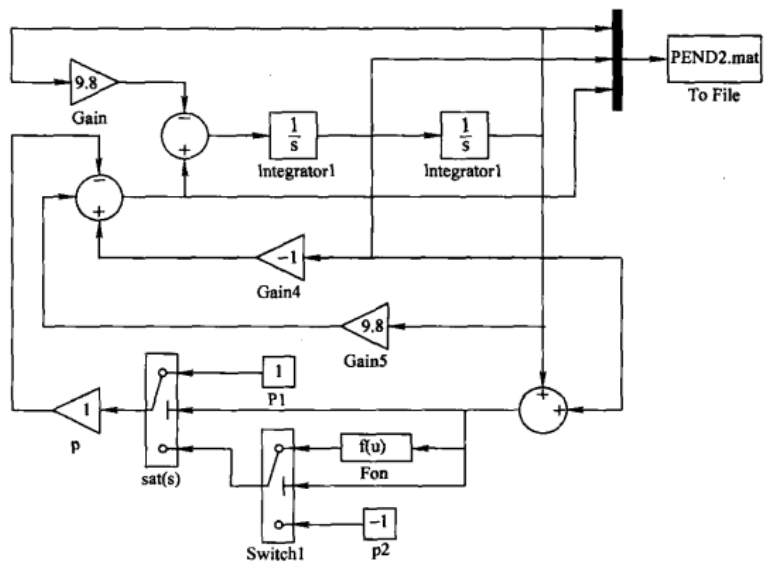


图 3-58　柔化 sgn(s) 函数后的系统仿真模型

$$\text{sat}(s) = \begin{cases} 1 & s > \dfrac{\pi}{\sqrt{2}}\varepsilon \\ \sin\left(\dfrac{s}{\sqrt{2}\,\varepsilon}\right) & |s| \leqslant \dfrac{\pi}{\sqrt{2}}\varepsilon \\ -1 & s < -\dfrac{\pi}{\sqrt{2}}\varepsilon \end{cases}$$

式中，ε 为大于零的正数，且 $\Delta = \dfrac{\pi}{\sqrt{2}}\varepsilon$。当 ε 取无穷小时，sat(s) 函数便非常逼近 sgn(s) 函数。采用 sat(s) 函数来代替 sgn(s) 函数后的系统仿真模型如图 3-58 所示，系统仿真结果如图 3-59 所示。

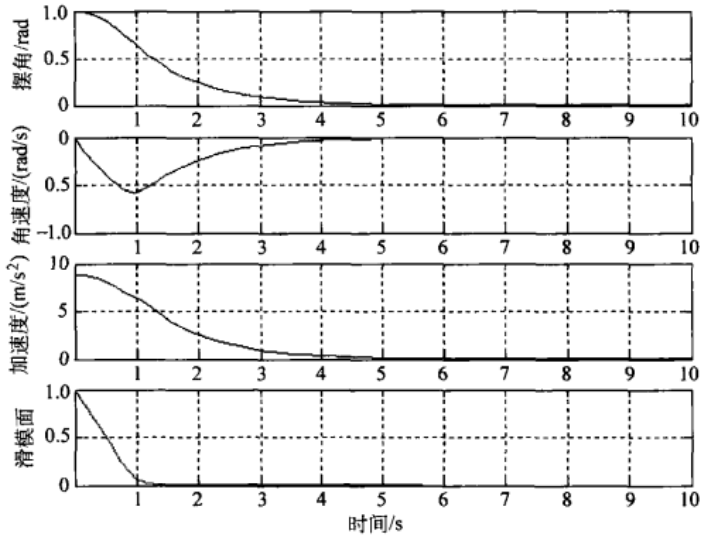


图 3-59 采用连续函数后的系统时域图

通过以上仿真可以看出，采用连续函数 sat(s) 代替不连续函数 sign(s) 后，系统仿真速度加快，不影响系统控制效果，且加速度不存在抖振现象，易于工程实现，能满足我们所期望的结果。

针对上述所提出的四种提高仿真效率的解决策略，综合比较其性能，见表 3-4。

表 3-4 四种提高仿真速度策略的性能比较

指标 / 方法	过零检测	步长	算法选择	连续性	有无抖振	仿真速度	仿真精度	实用性	推荐等级
Fcn 函数模块法	on	变	ODE45	不连续	有	较快	较低	较差	Ⅲ
无过零检测法	off	变	ODE45	不连续	有	较快	较高	较好	Ⅱ
fixed-step 求解器法	on	定	ODE1	不连续	有	快	低	差	Ⅳ
柔化 sgn(s) 函数法	on	变	ODE45	连续	无	快	高	好	Ⅰ

因此，在控制算法中，如果出现了像 signum 这样一类的非连续性模块而影响系统仿真速度和效率，建议采用以下方法处理：

1）最好将 signum 函数进行连续化，这样可以使系统仿真速度快且仿真效果好，无抖振，易于工程实现。

2）可以直接采用 fixed-step 法或者无过零检测法，采用这种方法，仿真速度快。但是这种方法可能忽略一些重要的过零信息，同时由于仍然存在非连续性问题而导致系统抖振的存在。

3）也可以采用 Fcn 函数模块，将运动学方程写成符合 C 语言规范的表达式，但是由于 Fcn 模块不支持过零，结果一些不连续的拐角点被漏掉了。所以在精度要求较高的场合，不宜采用 Fcn 函数模块。

三、几点讨论

1）你了解 Simulink 仿真运行的原理吗？

2）常用的 Simulink 仿真算法有哪些？分别适用于解决什么问题？

3）有哪些环节/过程影响 Simulink 仿真的效率？有何解决策略？

小　结

本章从控制系统常见的结构形式入手，叙述了对系统结构的拓扑描述方式，由此引出了面向连续和离散（采样）控制系统结构图的计算机仿真方法及其程序实现。

1）控制系统结构形式有多种，常见的如单输入-单输出开环控制、前馈控制及闭环（反馈）控制和多输入-多输出控制等，用图论拓扑结构加以描述，会给控制系统数字仿真带来极大方便。

2）任何复杂连接结构的线性控制系统都是由一些简单的线性环节组合而成的，按照它们之间相互连接的拓扑关系列出连接矩阵，可以得到能清晰简洁地描述复杂连接系统的“仿真模型”。

3）具有典型闭环结构形式的系统，“二次模型化”过程简单明了。在仿真模型基础上，采用 MATLAB 语言，可以很方便地编程实现数值积分法求解系统动态响应，但无法观察系统中间变量的变化规律，也难以分析非线性环节的影响。

4）具有复杂连接闭环结构形式的系统，可用式（3-5）形式的一阶环节作为典型环节，再运用拓扑描述中联接矩阵的表达方法而得到此类系统结构仿真模型，然后再通过数值积分法求取各环节动态响应。但必须注意，所确定的典型环节中，参数 $B_i \neq 0$，以保证系统仿真求解的基本条件。

5）离散相似法是一种对连续系统通过按系统离散化（或按环节离散化）方法来求解系统动态响应，可使仿真运算求解过程大为简化，避免了繁琐的数值积分求解过程，并能很方便地考虑常见非线性环节（饱和、死区、滞环等）的影响，从而编程实现含有典型非线性环节的控制系统的仿真求解。

但离散相似法对计算步长 h 选取要求较严格，一般欲达到数值积分法的相应精度，应取比其小一个数量级的 h 值，否则，会因 h 取得不当而引起计算结果数值不稳定，失去仿真意

义。

6）计算机控制系统实际上是一种离散（采样）控制系统，其特点是系统中具有不随时间连续变化的物理量。

采样控制系统的数学描述一般采用 z 变换、扩展 z 变换、差分方程和 z 传递函数等形式，应根据对采样控制系统不同仿真要求而确定；采样系统数学模型也可以应用 MATLAB 语言进行相互转换。

7）对采样控制系统的仿真应根据采样控制系统的结构形式不同而选用相应的方法。采样控制系统各环节均可表示为 z 传递函数形式（如控制器 $D(z)$，被控制对象 $G_0(z)$ 等），直接用差分方程递推求解。

采样控制系统中包含变量连续变化的控制器 $D(s)$ 或被控制对象 $G_0(s)$，应采用双重循环方式。即在内环，两采样周期 kT 和 $(k+1)T$ 之间，对连续部分采用按环节离散化方法，以计算步长 h 求取相应环节的连续变化响应值；而在外环，对整个系统则以系统采样周期 T 为步长，计算所有离散（采样）环节和连续环节在离散时刻 $t=kT$ 的离散变化响应值。

8）采样控制系统高阶差分方程的仿真求解步骤是“存入→平移→取出→计算”。

采样控制系统中含有纯滞后环节的仿真，可以根据纯滞后时间 τ 是否为采样周期 T 的整倍数，对两采样时刻之间值采用直线插补运算求解，或用扩展 z 变换处理方式运算求解。

习　题

3-1　思考题：

1）控制系统的结构形式用图论中的拓扑描述有何优点？

2）连接矩阵的作用是什么？阵中元素 w_{ij} 有几种形式，分别表示什么意义？

3）典型一阶环节式（3-5）中，若 $B_i=0$，会造成什么问题？

4）试从系统稳定性角度阐述按环节离散化（离散相似法）仿真步长 h 对系统动态响应数值解的影响。

5）为什么称计算机控制系统为采样控制系统或离散控制系统？其仿真有何特点？

3-2　设典型闭环结构控制系统如图 3-60 所示，当阶跃输入幅值 $R=20$ 时，用 sp3_1. m 求取输出 $y(t)$ 的响应。

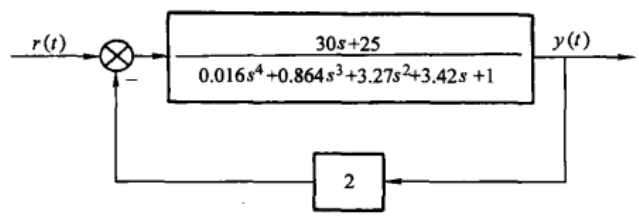


图 3-60　题 3-2 图

3-3　试说明系统中纯比例环节、纯微分环节对式（3-10）中 $\boldsymbol{Q}$ 阵的逆的存在有何影响。

3-4　系统结构图如图 3-61，写出该系统的连接矩阵 $\boldsymbol{W}$ 和 $\boldsymbol{W}_0$，并写出连接矩阵非零元素阵 $\boldsymbol{W}_{IJ}$。

3-5　图 3-61 中，若各环节传递函数已知为

$$G_1(s)=\frac{1}{1+0.01s};\ G_2(s)=\frac{1+0.17s}{0.085s};\ G_3(s)=\frac{1}{1+0.01s};$$

$$G_4(s)=\frac{1+0.15s}{0.051s};\ G_5(s)=\frac{70}{1+0.0067s};\ G_6(s)=\frac{0.21}{1+0.15s};$$

$$G_7(s)=\frac{130}{s};\ G_8(s)=\frac{0.1}{1+0.01s};\ G_9(s)=\frac{0.0044}{1+0.01s};$$

但 $G_{10}(s)=0.212$；重新列写连接矩阵 $\boldsymbol{W}$、$\boldsymbol{W}_0$ 和非零元素阵 $\boldsymbol{W}_{IJ}$，将程序 sp3_2. m 完善后，应用 sp3_2. m 求输出 y_7 的响应曲线。

3-6　若系统为图 3-5b 双输入-双输出结构，试写出该系统的连接矩阵 $\boldsymbol{W}$、$\boldsymbol{W}_0$，说明应注意什么？

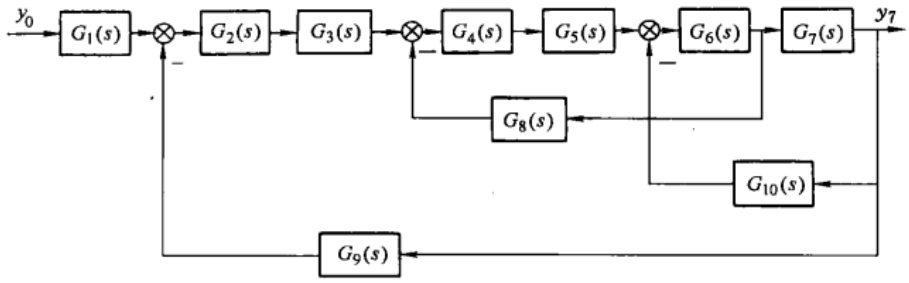


图 3-61　题 3-4 图

3-7　用离散相似法仿真程序 sp3_4. m 重求题 3-5 输出 y_7 的数据与曲线，并与四阶龙格-库塔法比较精度。

3-8　求图 3-62 非线性系统的输出响应 $y(t)$，并与无非线性环节情况进行比较。

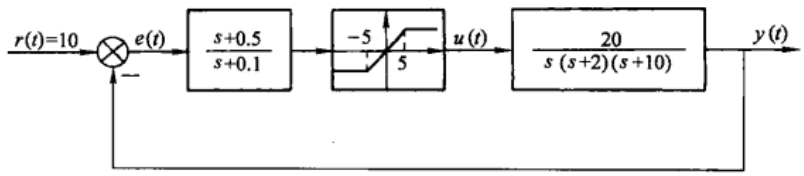


图 3-62　题 3-8 图

3-9　以 MATLAB 语言函数文件形式编写如图 3-25 带滞环三位继电特性非线性环节的仿真程序。（提示：非线性特性由两个非线性参数 s_1、s_2 描述）

3-10　采样控制系统如图 3-63 所示，编写程序实现对该系统的仿真分析（提示：连续部分按环节离散化考虑）。图中

$$D(z)=\frac{U(z)}{E(z)}=\frac{K_P\left[\left(1+\frac{T}{T_I}+\frac{T_D}{T}\right)-\left(1+2\frac{T_D}{T}\right)z^{-1}+\frac{T_D}{T}z^{-2}\right]}{1-z^{-1}}$$

为典型数字 PID 控制器；$K_P=0.65$ 为比例系数；$T_I=0.7$ 为积分时间常数；$T_D=0.2$ 为微分时间常数；

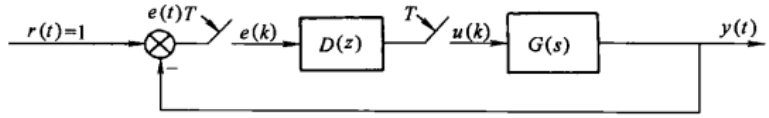


图 3-63　题 3-10 图

$$G(s)=\frac{Y(s)}{U(s)}=\frac{e^{-\tau_3 s}}{(T_1 s+1)(T_2 s+1)}$$

为具有纯滞后特性的典型二阶控制对象；$T_1=0.3s$；$T_2=0.3s$；$T_3=0.4s$。

第四章　控制系统 CAD

第一节　概　　述

控制系统计算机辅助设计是一门以计算机为工具进行控制系统设计与分析的技术，简称控制系统 CAD。

20 世纪 50 年代，频域法在控制系统分析与设计中得到迅速的发展，而时域法相对处于停滞状态。其主要原因在于拉普拉斯变换将时域中的微分方程求解转化为频域中的代数运算，而工程师只能通过手工计算和一些图表的帮助来进行控制系统的粗略设计。在这一时期，为了得到复杂系统在时域中的解曾广泛采用了模拟计算机仿真的方法。模拟计算机的编程方便、运算并行、模块形象等优点，使得它在控制系统 CAD 的历史上占有重要地位。但是，由于数字计算机的迅速发展，模拟机在精度、柔性以及价格等方面的弱点终于无法与数字机抗衡，因此已逐渐被淘汰。

20 世纪 60 年代，数字计算机开始被应用于控制系统的计算机辅助设计，在数字仿真理论、数值算法以及各种应用程序的设计与开发等方面取得了许多建设性的成果。

20 世纪 70 年代，开始出现了控制系统计算机辅助设计的软件包，英国的 H. H. Rosenbrock 学派将线性单变量控制系统的频域理论推广到多变量系统，随后曼彻斯特（Manchester）大学的控制系统中心完成了该系统的计算机辅助设计软件包；日本的古田胜久主持开发的 DPACS-F 软件，在处理多变量系统的分析和设计上也很有特色。同时，国际自动控制领域的知名学者，瑞典隆德（Lund）大学的 K. J. Astrom 和他的学生用状态空间法发展了多变量系统控制理论，其在控制系统 CAD 软件结构上克服了刻板的提问与回答的限制，提出了命令式的人机交互界面，在控制系统 CAD 中给设计者以主动权，将“人机交互”技术提高到一个新阶段。他们先后完成了由 Idpac Intrac Modpac Polpac Simnon 和 Synpac 等六个软件组成的一整套软件系统，在国际上具有重要影响。

20 世纪 80 年代以来，在众多仿真语言和仿真软件包的基础上，美国的 Mathwork 公司推出了 MATLAB 软件系统，它具有模块化的计算方法，可视化与智能化的人机交互功能，丰富的矩阵运算、图形绘制、数据处理函数、基于模型化图形组态的动态系统仿真工具 Simulink 等优秀软件，而且 Mathwork 公司密切注意科技发展的最新成果，及时地与不同领域的知名专家合作，推出了许多控制系统 CAD 的“工具箱”，被人们誉为“巨人肩上的工具”。

现在，MATLAB 已从单纯的“矩阵实验室”渗透到科学与工程计算的许多领域，成为控制系统 CAD 领域最受欢迎的软件系统。

本章以 MATLAB 及 Simulink 为工具，就经典控制理论和现代控制理论中的一些具体问题进行回顾与讨论，从而使大家进一步体会“控制系统 CAD”技术在控制系统分析、设计及理论学习中的作用和意义。

第二节　经典控制理论 CAD

把用频率法研究单输入-单输出线性定常系统，用传递函数描述控制系统，用根轨迹、频率特性等试凑法设计和分析控制系统的理论与方法，称为经典控制理论。本节我们将讨论其中的固有特性分析、系统设计（校正）方法以及控制系统优化设计等问题。

一、控制系统固有特性分析

当控制系统的数学模型建立以后，就可以采用 CAD 的方法来分析系统自身的性能了，常用的分析方法有时域分析、频域分析和根轨迹三种方法。

（一）时域分析

利用时域分析能够了解控制系统的动态性能，这可以通过系统在输入信号作用下的过渡过程来评判。Simulink 非常适合于做系统的时域分析，下面举例说明之。

例 4-1　二阶系统的动态性能分析。

解　为分析方便，通常将二阶系统的闭环传递函数写成如下标准形式

$$\phi(s) = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}$$

式中，ξ 为阻尼比；ω_n 为无阻尼自振角频率。随着阻尼比的不同，系统闭环极点的位置也不同，它可分成以下四种情况：①欠阻尼，即 $0 < \xi < 1$；②临界阻尼，即 $\xi = 1$；③过阻尼，即 $\xi > 1$；④无阻尼，即 $\xi = 0$。

分别取 $\xi = 0$，0.4，0.8，1.0，1.4，$\omega_n = 1.0$，二阶系统单位阶跃响应的图形组态及仿真结果如图 4-1 所示。

由图中的仿真结果可以看出，二阶系统在单位阶越函数的作用下，随着阻尼比的减少，振荡程度越来越严重，当 $\xi = 0$ 时出现了等幅振荡。当 $\xi \geqslant 1$ 时，二阶系统的过渡过程具有单调上升的特性。

在欠阻尼（$0 < \xi < 1$）特性中，对应 $\xi = 0.4 \sim 0.8$ 时的过渡过程不仅具有比 $\xi = 1$ 时更短的响应时间，而且振荡程度也不是很严重。因此，通常希望二阶系统能够工作在 $\xi = 0.4 \sim 0.8$ 的欠阻尼状态。

（二）频域分析

以频率特性作为数学模型来分析、设计控制系统的方法称为频率特性法，它是系统频域分析的主要方法。

频率特性具有明确的物理意义，计算量较小，一般可采用作图的方法或实验的方法求出系统或元件的频率特性，这对于机理复杂或机理不明确而难以列写微分方程的系统或元件，具有重要的实用价值。也正是频率特性法的上述优点，使得它在工程技术领域得到广泛的应用。

实际应用时，由于频率特性 $G(j\omega)$ 的代数表达式较复杂，所以总是采用图形表示法，直观地表达 $G(j\omega)$ 的幅值与相角随频率变化的情况，从中可分析得出系统的静态与动态性能情况。最常用的频率特性图是对数坐标图（即伯德图）。MATLAB 语言中提供了绘制伯德

图的专用命令：bode，其表达格式有如下几种：

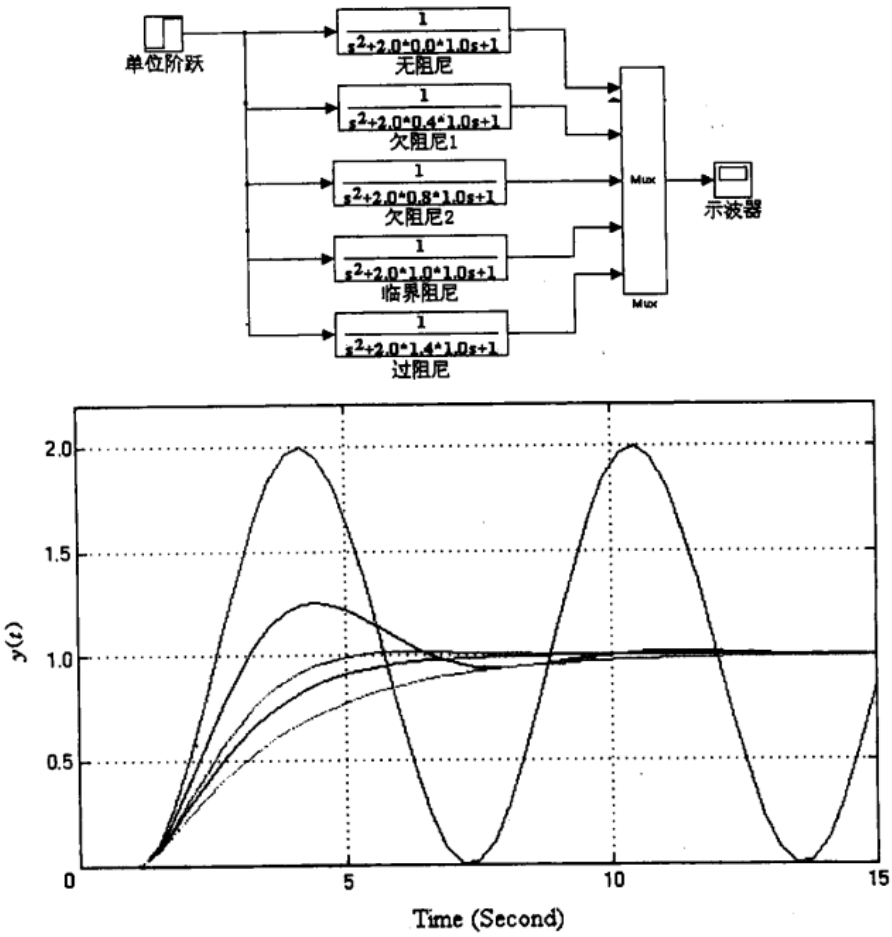


图 4-1 二阶系统单位阶跃响应的图形组态及仿真结果

>> bode（num，den）

>> [mag，phase，w] = bode（num，den）

>> [mag，phase] = bode（num，den，w）

其中，w 表示频率 ω；第一种命令在屏幕中的上下两部分分别生成幅频特性和相频特性；第二种命令可自动生成一行矢量的频率点；第三个命令可定义所需的频率范围。同时，要绘制伯德图还需下述命令：

>> subplot（2，1，1），semilogx（w，20 * log 10（mag））,

>> subplot（2，1，2），semilogx（w，phase），

其中，前者的第一个命令把屏幕分成两个部分，并把幅频特性放在屏幕的上半部；第二个命令（即 semilogx）可生成一个半对数坐标图（横轴是以 10 为底的对数值坐标轴，而纵轴则是以 dB 为单位表示的幅值）；后者之命令是将系统相频特性放置在屏幕的下半部分。若想以 Hz 为单位，可用 w/2 * pi 来代替 w。若要指定频率范围，可使用 logspace 命令：

>> w = logspace（m，n，npts）

其可生成一个以 10 为底的对数向量（$10^m \sim 10^n$），点数（npts）可任意选定。下面举例说明

伯德图的绘制过程。

例 4-2　已知某控制系统的开环传递函数为 $G(s)=\dfrac{K}{s(s+1)(s+2)}$，$K=1.5$，试绘制系统的开环频率特性曲线，即系统的伯德图。

解　下面是使用 logspace 命令生成具有 100 个频率点的对数坐标频率特性程序：

```
K = 1.5; ng = 1.0; dg = poly ( [0, -1, -2]);
w = logspace ( -1, 1, 100);
[m, p] = bode (K * ng, dg, w);
subplot (2, 1, 1); semilogx (w, 20 * log10 (m));
grid; ylabel ('Gain (dB)');
subplot (2, 1, 2); semilogx (w, p); grid;
xlabel ('Frequency (rad/s)'); ylabel ('Phase (deg)');
```

其执行结果如图 4-2 所示，从中可见，系统的幅值裕量和相角裕量分别为 10dB 与 45°。

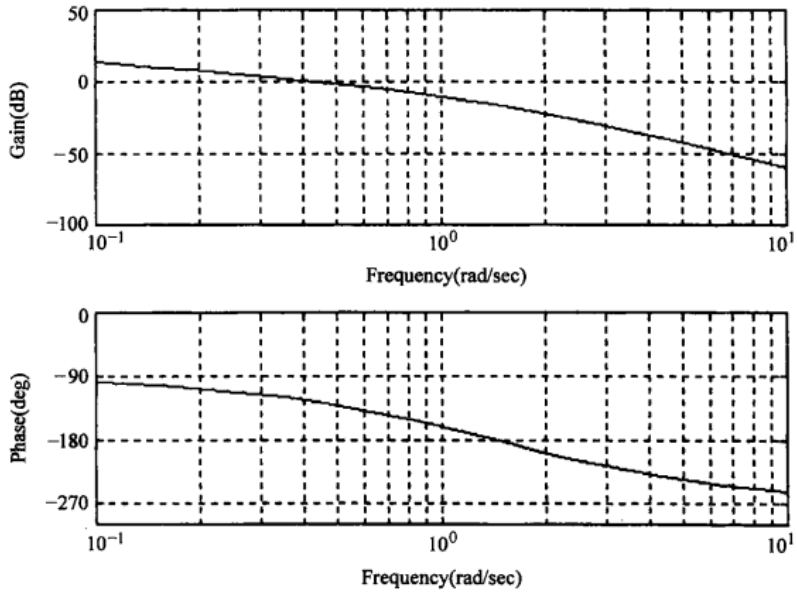


图 4-2　开环系统伯德图

此外，使用 nyquist 与 nichols 命令可以得到系统的另外两种频率响应特性，即复平面上的奈奎斯特图形与尼克尔斯图形。

（三）根轨迹

根轨迹是 W. R. Evans 提出的一种求解闭环系统特征根的非常简便的图解方法。由于控制系统的动态性能是由系统闭环零极点共同决定的，而控制系统的稳定性又是由闭环系统极点唯一确定的。因此，在分析控制系统动态性能时，确定闭环系统的零极点在 S 平面上的位置就显得特别重要。

MATLAB 语言提供了绘制单输入-单输出系统根轨迹的命令 rlocus，其基本形式如下：

```
>> rlocus (num, den)
>> rlocus (num, den, k)
```

执行该命令，根轨迹图会自动生成。如果 K 值给定，则将按照给定的参数绘制，否则增益

是自动确定的。

在系统的分析过程中，常常希望确定根轨迹上某一点处的增益值，MATLAB 为此提供了 rlocfind 命令。其首先要得到系统的根轨迹，然后执行如下命令：

```
>> [k, poles] = rlocfind (num, den)
```

执行命令后，将在屏幕上的图形中生成一个十字光标，使用鼠标器移动它至所希望的位置，然后敲击左键即可得到该极点的位置坐标值以及它所对应的增益 K 值。下面举例说明根轨迹的绘制过程。

例 4-3 已知某负反馈系统的开环传递函数为 $G(s)H(s) = \dfrac{K}{s(s+1)(s+2)}$，试绘制系统根轨迹。

解 执行语句：ng = 1.0; dg = poly ([0, -1, -2]); rlocus (ng, dg)，即可生成图 4-3 所示的系统根轨迹。从中可见，根轨迹的分支数是 3 个，三条根轨迹的起点分别是 (0, j0)，(-1, j0)，(-2, j0)，终点均为无穷远。

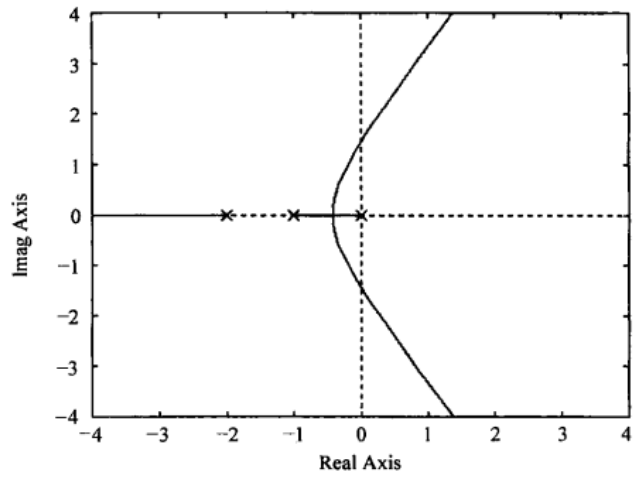


图 4-3 系统根轨迹图

二、控制系统的设计方法

所谓控制系统的设计就是在系统中引入适当的环节，用以对原有系统的某些性能（如上升时间、超调量、过渡过程时间等）进行校正（又称综合），使之达到理想的结果，故又称之为系统的校正与综合。对于采用传递函数描述的控制系统，常用的经典设计方法是根轨迹与频域法。下面介绍几种常用的系统校正方法的计算机辅助设计实现。

(一) 超前校正

控制系统中常用的校正装置是带有单一零点与一个极点的滤波器，其传递函数描述为 $K(s) = K_c \dfrac{s+a}{s+b}$。若其零点出现在极点之前（即 $0 < a < b$），则称 $K(s)$ 为超前校正，否则称之为滞后校正，其最大的相位补偿点为 $\omega = \sqrt{ab}$。

所有的校正装置都将影响闭环系统的动态性能。一般来讲，超前校正会使系统的相角裕量增加，从而提高系统的相对稳定性，致使闭环系统的频带扩宽，这也正是我们希望的。当然，系统频带的加宽也会带来一定的噪声信号，这也是系统所不希望的。

在频域法中，采用伯德图进行设计是最常见的，其基本思想是：改变原有系统开环频率特性的形状，使其具有希望的低频增益（满足稳态误差的要求）、希望的增益穿越频率（满足响应速度的要求）和充分的稳定裕量。在伯德图设计方法中，为方便起见常常采用如下形式的传递函数的校正装置

$$K(s) = K_c \frac{\alpha TS + 1}{TS + 1}$$

下面举例说明系统的超前校正方法。

例 4-4　已知被控对象的传递函数为 $G(s)=\dfrac{400}{s(s^2+30s+200)}$，系统的设计指标要求如下：①速度误差常数为 10；②相角裕量为 45°。

解　由速度误差常数的要求可求得 $K_c=5$，则可绘出 $K_cG(j\omega)$ 的伯德图如图 4-4 所示，从中可见，此时系统相角的稳定裕量大约为 32°，因此需要再补偿 13°。

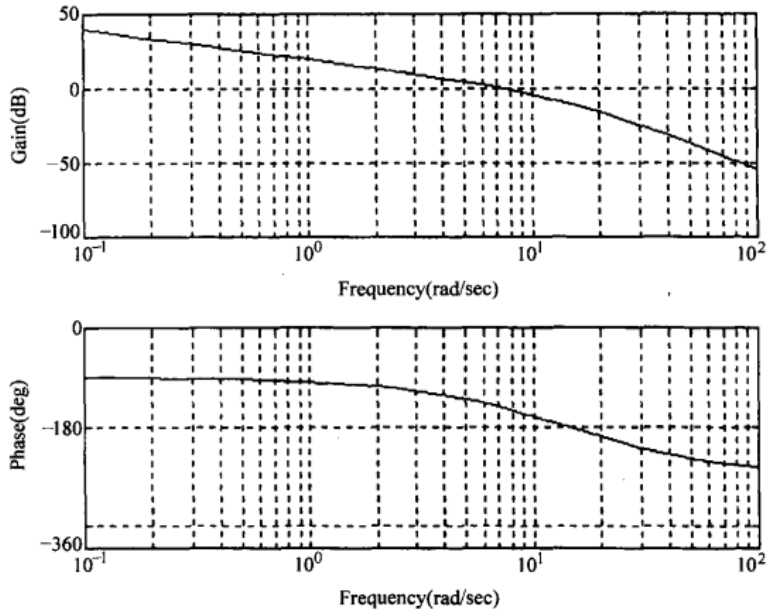


图 4-4　$K_cG(j\omega)$ 的伯德图

在计算 α 时，应再将 ϕ 加上 5°的裕量，则可得 $\alpha=1.89$。由此可进一步求得：$-10\log\alpha=-2.77\text{dB}$，$\omega_{gc}=9\text{rad/s}$，$T=\dfrac{1}{\sqrt{\alpha}\omega_{gc}}=0.08\text{s}$。由上得校正装置传递函数为

$$K(s)=5\times\frac{0.15s+1}{0.08s+1}$$

不难求得校正后的相角裕量为 41°，接近系统设计要求。校正前后系统的模型及动态仿真结果如图 4-5 所示，从中可见，校正后系统的动态响应速度明显加快，这与理论分析结果是相符的。

（二）滞后校正

在上述校正装置中若零点出现在极点之后（即 $a>b>0$），则 $K(s)$ 即为滞后校正。由于滞后校正装置给系统加入了滞后的相角，因而将会使得系统的动态稳定性变差。如果原有系统的稳定性已经较差（相角裕量较小），则系统校正中不宜采用滞后校正方法。

滞后校正可降低系统稳态误差。这一点可以假想校正装置在极点非常小（趋于零）的情况下，滞后校正装置将近似于一个积分器，它使得原系统增型，因而可以降低系统的稳态误差。

滞后校正将使得闭环系统的带宽降低（ω_{gc}减小），从而使系统的动态响应速度变慢，这

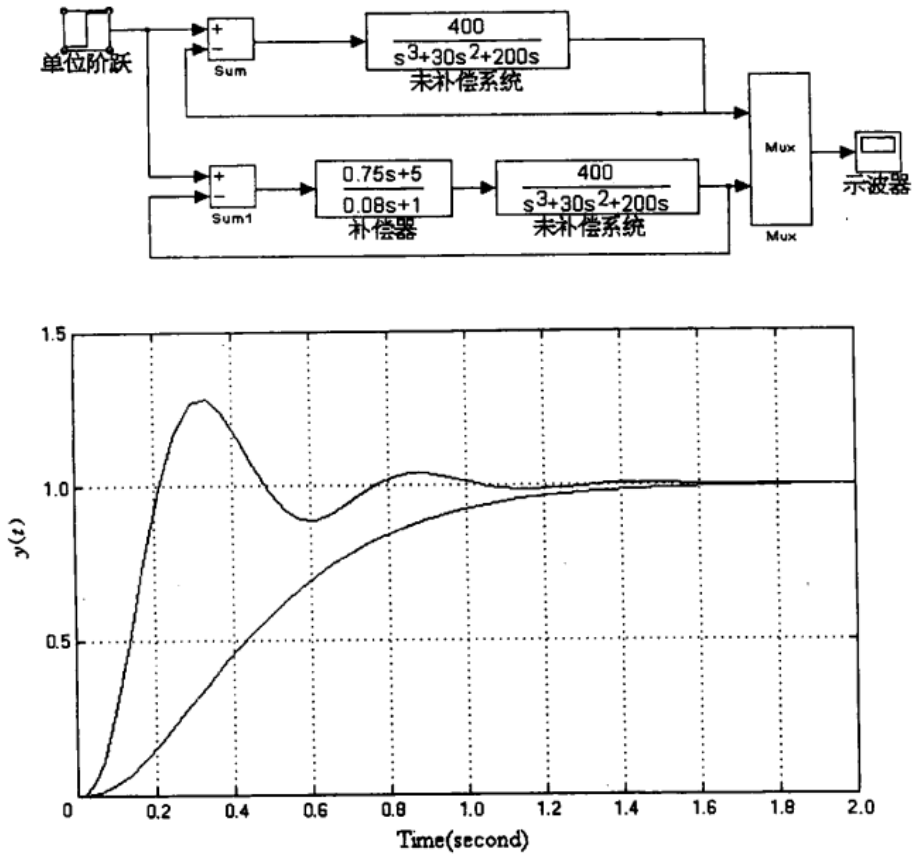


图 4-5 系统图形组态及仿真结果

有利于减小外部噪声信号对系统的影响。

下面的例子说明了滞后校正的实现过程。

例 4-5 已知单位负反馈系统固有部分的传递函数为 $G_0(s)=\dfrac{K}{s(s+1)(0.5s+1)}$，若要求系统满足如下性能指标：开环放大倍数 $K_v=5$，相角裕量 $r(\omega_c)\geqslant 40°$，幅值裕量 $K_g\geqslant 10\text{dB}$，试设计校正装置。

解 原有系统的伯德图如图 4-6 所示，从中可见，当 $\omega=1$ 时，$20\log K_v=20\log 5=14\text{dB}$，$\omega_{c0}=2.1\text{rad/s}$，相角裕量 $r_0=-20°$（系统不稳定）。

由所要求的校正后 $r(\omega_c)\geqslant 40°$，可以解析求得：$\alpha=0.1$，$T=100\text{s}$，则可得出滞后校正装置为

$$G_c(s)=\frac{10s+1}{100s+1}$$

校正后系统的开环传递函数为

$$G(s)=G_c(s)G_0(s)=\frac{5(10s+1)}{s(s+1)(0.5s+1)(100s+1)}$$

其伯德图如图 4-7 所示，系统图形组态及仿真结果如图 4-8 所示。从中可见，校正后系统具

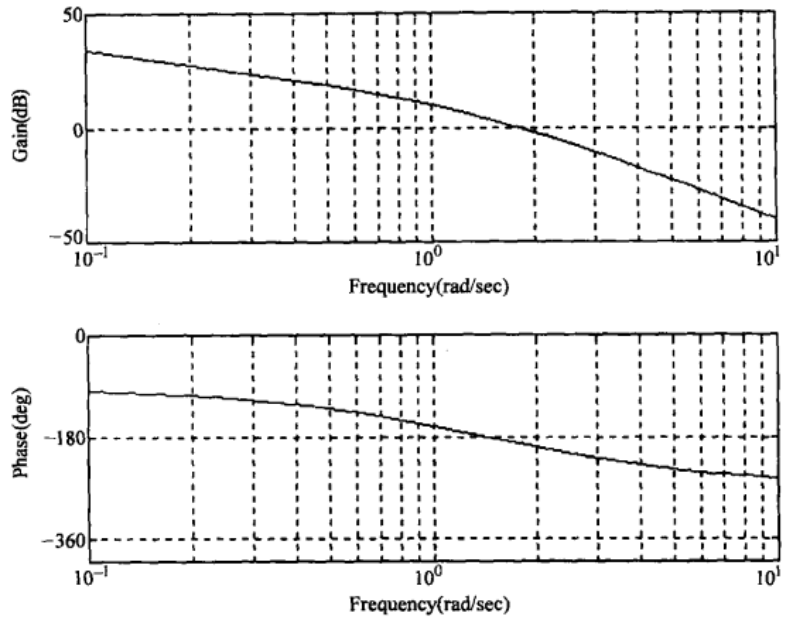


图 4-6　原有系统伯德图

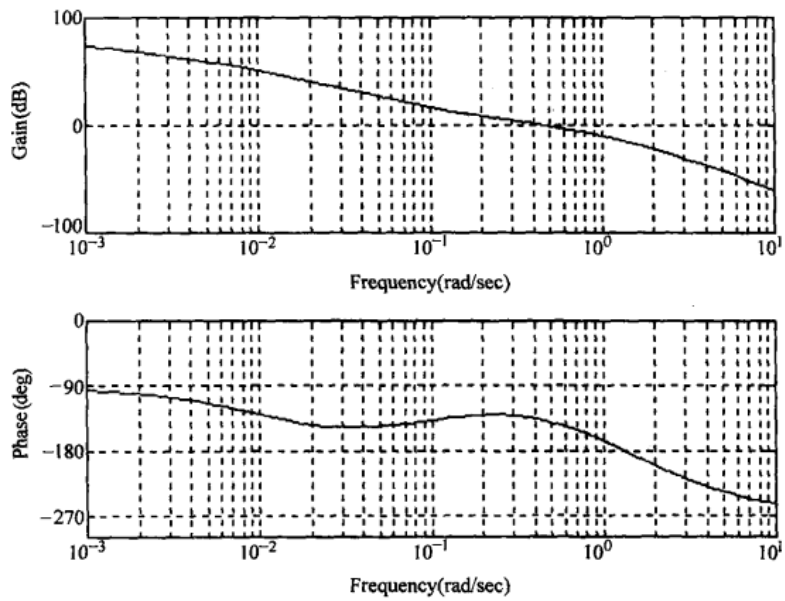


图 4-7　校正后系统伯德图

有良好的动态响应过程，而校正前系统动态是不稳定的。

（三）滞后-超前校正

超前校正是用超前相角对系统实现校正的，其优点是可改善系统的动态性能（加宽系统频带，提高了系统的响应速度），但同时它一方面使系统放大倍数有所衰减，不利于改善系统的稳态性能；另一方面系统频带的增加也降低了系统的抗干扰能力。而滞后校正是用高频段的衰减特性对系统进行校正，其作用是改善系统的稳态性能，但同时它一方面引进了滞后相位而对改善系统的动态性能不利；另一方面系统的频带有所减小，降低了系统的动态响

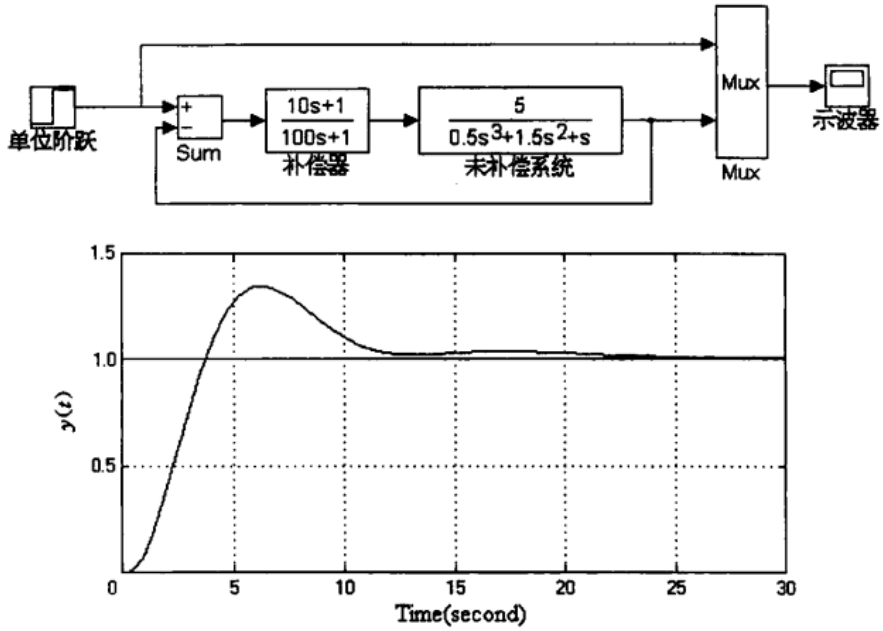


图 4-8 系统图形组态及仿真结果

应速度，但增加了系统的抗干扰能力。

从以上分析可见，若同时引入超前校正和滞后校正，可同时改善系统的动态性能与稳态性能，两者的优点得到发挥，而缺点又可以相互补偿。因此，在实际工作中滞后-超前校正也常被人们所采用。下面通过具体例子说明其实现过程。

例 4-6 已知单位负反馈系统固有部分的传递函数为 $G_0(s)=\dfrac{K}{s(s+1)(0.5s+1)}$，要求系统满足下述指标：开环放大倍数 $K_v=10$，相角裕量 $r(\omega_c)\geqslant 40°$，幅值裕量 $K_g\geqslant 10\text{dB}$。

解 首先根据所要求的开环放大倍数绘制系统的伯德图（略），从中不难求得未校正系统的相角裕度为 $-33°$，而 $\omega_g=1.4\text{rad/s}$，$K_g<0\text{dB}$。所以，此时系统是不稳定的，需要进行校正。

首先，为了保证响应速度，校正后的剪切频率不应离校正前的剪切频率太远，取 $\omega_c=1.4\text{rad/s}$，则此时有 $\angle G_0(j\omega_c)=-180°, r_0(\omega_c)=0°$，这样所需达到的相角裕量 $r(\omega_c)\geqslant 40°$ 就完全由滞后校正环节给出。根据 $\dfrac{1}{T_2}=\left(\dfrac{1}{5}\sim\dfrac{1}{10}\right)\omega_c$ 原则，即可确定滞后校正参数，其中还需考虑到滞后校正环节部分在校正后的剪切频率 ω_c 处的相角不能小于 $-5°$，以及所设计校正环节的实现问题。在这里，取 $\dfrac{1}{T_2}=\dfrac{\omega_c}{10}=\dfrac{1.4}{10}=0.14\text{rad/s}$，则 $T_2=\dfrac{1}{0.14}\text{s}=7.14\text{s}$；再取 $\beta=10$，$\dfrac{1}{\beta T_2}=0.014\text{rad/s}$，$\beta T_2=71.4\text{s}$，则所设计的滞后校正装置的传递函数为 $G_{c1}(s)=\dfrac{7.14s+1}{71.4s+1}$。

其次，确定超前校正环节参数的原则是要保证校正后系统剪切频率为 1.4rad/s。同理，我们可解得如下的超前校正装置的传递函数为 $G_{c2}(s)=\dfrac{1.43s+1}{0.143s+1}$。从上可得滞后-超前校正

环节之传递函数为

$$G_c(s) = G_{c1}(s)G_{c2}(s) = \frac{(1.43s+1)(7.14s+1)}{(0.143s+1)(71.4s+1)}$$

校正后闭环系统的单位阶越响应及其图形组态如图 4-9 所示，从中可见，校正后系统具有良好的动态响应，比原系统的性能有明显改善。

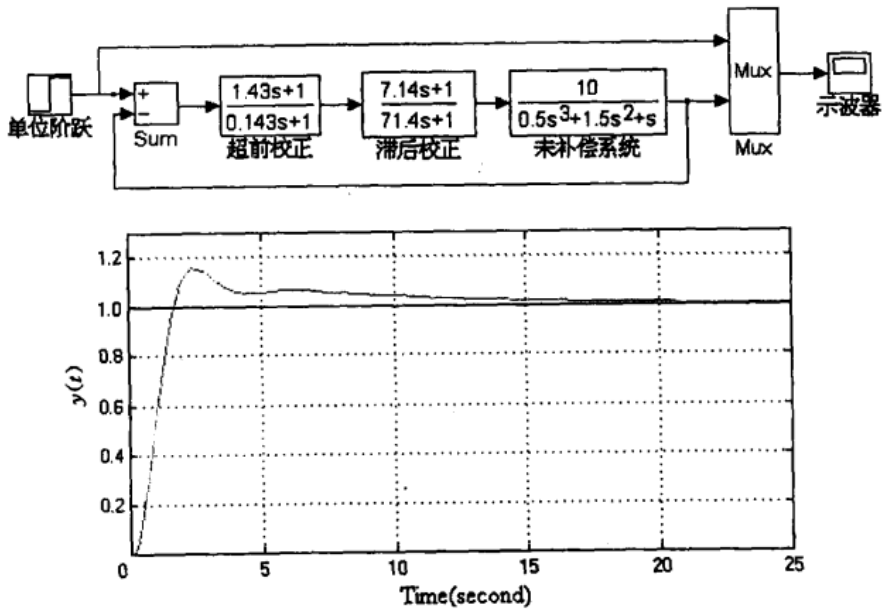


图 4-9　系统图形组态及仿真结果

（四）反馈校正

改善控制系统的性能，除采用上述三种串联校正方案以外，反馈校正也是广泛采用的系统设计方法之一。

对于前向通道上传递函数为 $G_0(s)$ 的单位闭环负反馈控制系统，所谓反馈校正就是在反馈通道上设置一校正装置 $H_c(s)$，在满足 $|G_0(j\omega)H_c(j\omega)| \gg 1$ 的条件下，则在我们感兴趣的频段里，就可以用 $\frac{1}{H_c(j\omega)}$ 取代原有闭环系统的特性，进而消除 $G_0(s)$ 中参数变化对系统性能的影响。当然 $H_c(s)$ 中的参数要有一定的稳定性和精度。通常称上述感兴趣的频段为接受校正频段。

控制系统采用反馈校正，除了能够获得与串联校正相同的效果外，还可赋予控制系统一些有利于改善系统控制性能的特殊功能。例如，比例负反馈可以减小其所包围环节的惯性，从而扩展系统频带；可以减小原有系统参数变化对系统性能的影响；可以消除系统不可变部分中不希望的特性等。

下面通过一具体例子说明反馈校正的实现过程。

例 4-7　已知某位置随动系统动态结构如图 4-10 所示，若要求满足如下性能指标：①开环放大倍数 $K_v = 100$；②超调量 $\delta \leqslant 23\%$；③过渡过程时间 $t_s \leqslant 0.6\text{s}$。试设计反馈校正装置 $H_c(s)$。

解　为便于确定 $H_c(s)$ 的参数，将图 4-10 等效成图 4-11 所示形式，并令 $sH_c(s) = H'_c(s)$，首先设计 $H'_c(s)$，然后再确定 $H_c(s)$ 的参数。

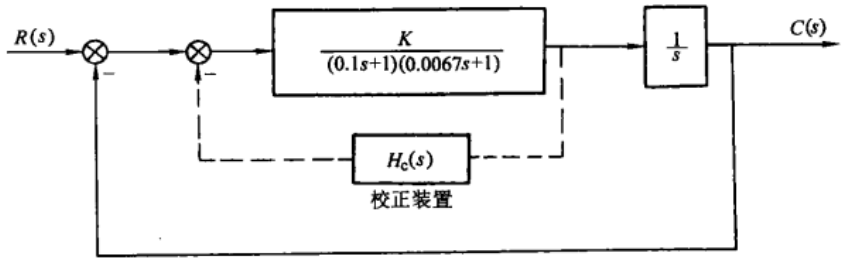


图 4-10 位置随动系统动态结构图

1）按要求的开环放大倍数 K，绘制不可变部分的伯德图，求未校正时的相角裕度和幅值裕度。

此时，不可变部分的传递函数为

$$G_0(s) = \frac{100}{s(0.1s+1)(0.0067s+1)}$$

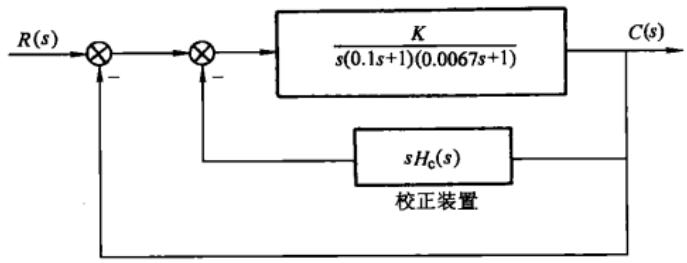


图 4-11 等效结构图

由其伯德图（略），可求得 $\omega_{c0} =$ 31rad/s。进而，可经计算求得 $r_0(\omega_{c0}) = 6.3°, \omega_g = \sqrt{\frac{1}{0.1 \times 0.0067}}\text{rad/s} = 38.6\text{rad/s}$。可见，原系统是稳定的，但其动态指标不符合要求，需进行校正处理。

2）求取期望频率特性。对于给定的时域指标 δ、t_s 可转成相应的频域指标，即由

$$\begin{cases} M_r = 0.6 + 2.5\delta = 0.6 + 2.5 \times 0.23 = 1.175 \\ \sin r(\omega_c) = \dfrac{1}{M_r} \end{cases}$$

可求得 $r(\omega_c) = 58°$，由

$$\begin{cases} k = 2 + 1.5(M_r - 1) + 2.5(M_r - 1)^2 = 2.34 \\ \omega_c = \dfrac{k\pi}{t_s} = \dfrac{2.34 \times 3.14}{0.6}\text{rad/s} = 12.246\text{rad/s} \end{cases}$$

可近似取 $\omega_c = 12\text{rad/s}$，由

$$\begin{cases} h = \dfrac{M_r + 1}{M_r - 1} = 12.4 \approx 12 \\ \omega_3 \geqslant \dfrac{2h}{h+1} \cdot \omega_c = 22\text{rad/s} \\ \omega_2 \leqslant \dfrac{\omega_3}{h} = 6.9\text{rad/s} \end{cases}$$

取 $\omega_3 = 83\text{rad/s}$，$\omega_2 = 5\text{rad/s}$。

另外，低频段的转折频率 ω_1 可由几何法求取为 $\omega_1 = 0.6\text{rad/s}$。这样即可确定校正后系统的开环传递函数为

$$G_c(s) = \frac{100}{s\left(\frac{1}{0.6}s + 1\right)(0.0067s + 1)} \frac{\left(\frac{1}{5}s + 1\right)}{\left(\frac{1}{83}s + 1\right)}$$

3）校验性能指标。采用传统的计算方法不难验证：校正后系统的 $r(\omega_c)=57.5°$，$K_v=100s^{-1}$，δ 与 t_s 也均满足设计要求。图 4-12 给出了校正前/后系统的图形组态及动态仿真结果，从中可见，校正后的系统动态性能明显得到改善，超调量 $\delta<23\%$，但过渡过程时间大于 0.6s，约为 0.8s，这是传统设计方法中存在一定误差所造成的，实际中可再适当作一些微调，以使指标达到要求。

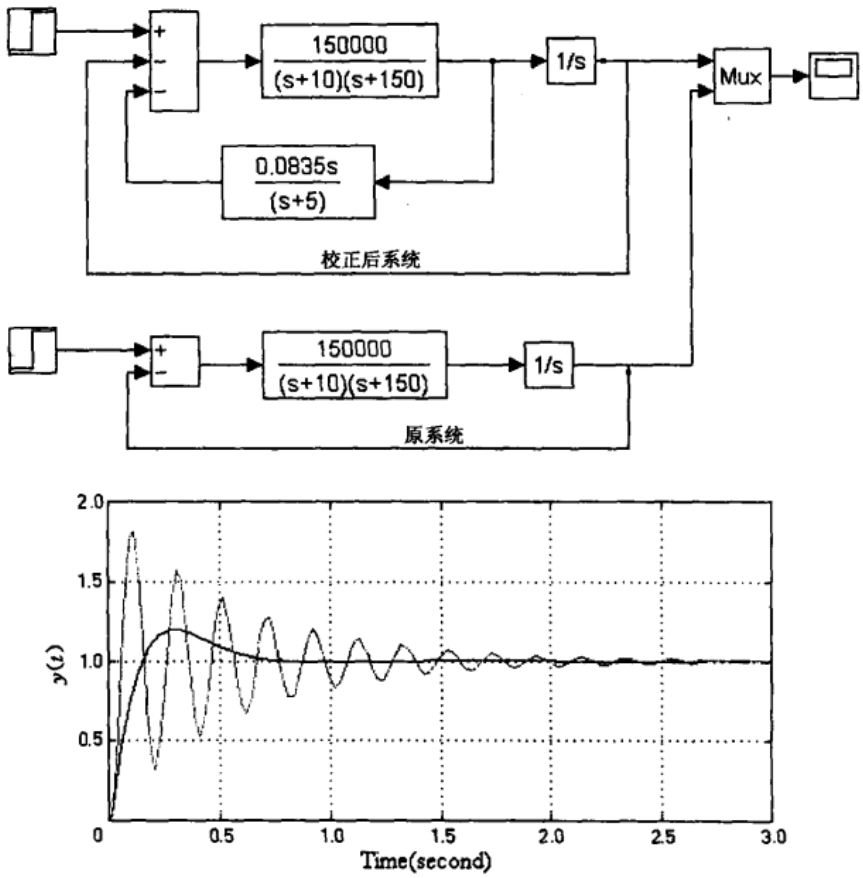


图 4-12　系统图形组态及仿真结果

4）确定校正装置的 $H_c(s)$ 参数。因为低于 ω_1 和高于 ω_3 的频段不需进行校正，所以只需考虑 ω_1（$=0.6$rad/s）$\sim\omega_3$（$=83$rad/s）的校正频段。由原系统频率特性及校正后系统的希望特性可推得：$H_c'(s)=\dfrac{K_n s^2}{\frac{1}{5}s+1}=\dfrac{K_n s^2}{0.2s+1}$，$K_n=\dfrac{1}{K_v\omega_1}=0.0167$，进而求得反馈校正装置的传递函数为

$$H_c(s)=\frac{H_c'(s)}{s}=\frac{0.0167s}{0.2s+1}$$

（五）设计方法小结

本节所述的设计方法属于经典控制理论的范畴，其实质上是用试探的方法来研究单输入-单输出线性定常系统的设计问题，因此其设计方案并不是绝对唯一的。概括起来有如下几点：

1）超前校正是利用超前校正环节（PD 或近似 PD 的控制规律）所提供的超前相角，增加校正后系统的相角裕度，并改变系统伯德图的中频段，使剪切频率增加，从而拓宽了系统的频带，提高了系统的快速性与相对稳定性。

2）滞后校正是利用滞后校正环节（或近似 PI 的控制规律）较小的转折频率来改变系统的中低频段特性，使固有部分的伯德图下移，剪切频率变小，频带宽度减小，利用固有部分所提供的相角裕度满足稳定裕度的设计要求，因此其适用于对快速性要求不高的系统。滞后校正可以在增加系统开环放大倍数的前提下保持系统的相对稳定性，在效果上可以提高系统的稳态精度。

3）滞后-超前校正综合了两种校正方法的优缺点，其既可有效地提高系统的动态响应速度，又可提高系统的稳态精度。

4）反馈校正是在中频段按期望特性来设计反馈环节，以在这一可校正频段上使闭环系统的频率特性主要由反馈环节的特性决定。因此，其可以消除中频段中不需要的特性并可有效抑制参数变化对系统性能的影响。

5）串联校正与反馈校正比较，前者设计容易、结构简单、易于实现且成本较低，而后者设计略显复杂，实现起来成本较高，但其校正后系统的性能相对较好。

总之，在控制系统设计中究竟采用哪种校正形式，在某种程度上取决于具体系统的结构及对系统的要求和被控对象的性质，也可应用专门化的 CAD 软件进行设计与分析。

三、控制系统的优化设计

所谓优化设计就是在所有可能的设计方案中寻找具有最优目标（或结果）的设计方法。它以一定的数学原理为依据，借助于数字计算机强大的分析计算能力，在自动控制、机械设计、经济管理和系统工程等方面为人们所广泛应用。

控制系统的优化设计包括两方面的内容：一方面是控制系统参数的最优化问题，即在系统构成确定的情况下选择适当的参数（对于非线性、时变系统，传统设计方法是难于实现的），以使系统的某种性能达到最佳；另一方面是系统控制器结构的最优化问题，即在系统控制对象确定的情况下选择适当的控制结构（或控制规律），以使系统的某种性能达到最佳。

本小节只讨论控制系统参数的优化设计问题。

（一）优化设计中的几个概念

一般情况下，由于优化设计是相对某些具体设计要求或某一人为规定的优化指标来寻优的，所以优化设计所得结果往往是相对的最优方案。图 4-13 给出了优化设计的流程框图，下面简要介绍其中的几个概念。

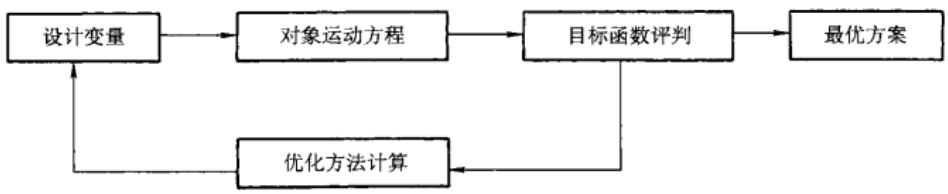


图 4-13 优化设计框图

（1）设计变量 在优化设计中，将某些有待选择的量值称为设计变量（如系统参数）。

通常，设计变量的初始值（可任意设定）不影响优化的结果，但影响优化设计的效率（计算时间）。

（2）约束条件　在优化设计中，某些设计变量的结果可能超出了某些设计要求的限制（不满足工程技术的要求）；计算机应能自动抛弃不合理的设计方案，而去继续寻找最优化方案，以提高优化设计的效果。这些限制条件在数学上称之为约束条件。

（3）目标函数　实际上，真正意义上的“最优”是很难寻求的，我们所说的“最优”是指在一定“条件”下的最优，目标函数就是人们设计的一个“条件”。目标函数的选择是整个优化设计过程中的最重要的决策之一，其选择得如何将直接影响最终结果。

（4）目标函数值的评定与权函数（罚函数）　在优化设计中，往往有几种方案可以被选择，如果各方案都满足约束条件的话，则目标函数值最小的即为最优方案。但是，在一个优化设计问题中，约束条件往往有好几项，而各种方案所满足的约束条件往往不一致，在这种情况下，怎样评判各个方案的优劣呢？这就需要把几个约束条件统一起来考虑再进行评判，在数学上有如下的综合目标函数

$$OBJ(\alpha) = OBJ_0(\alpha) + \sum_{i=1}^{m} C_i g_i^2(\alpha)$$

式中，$OBJ_0(\alpha)$ 为不考虑约束的目标函数；$g_i(\alpha)$，$i=0$，1，…m，为 m 个约束条件；C_i 为大于零的数，表示第 i 个约束在设计中所占的比重，称为权，$\sum_{i=1}^{m} C_i g_i^2(\alpha)$ 为权函数。

（二）优化设计原理——单纯形法

优化设计就是要寻找一组最优的设计变量以使目标函数取值最小（或最大），所以从数学上讲就是求取函数的极值问题，在工程上称之为“寻优问题”。但是，在工程上的一些实际问题中很难列写出其函数表示关系，因此常规由函数导数寻优的方法（梯度法）在许多工程问题上（如非线性时变控制系统中参数的优化）是不适用的。

在经典的优化设计方法中还有一类不必计算目标函数梯度的直接搜索方法，称之为随机试验法（或探索法）。大家可能都有这样的直观感觉，若能计算出（或测试出）若干点（即设计变量）处的函数值，然后将它们进行比较，从它们之间的大小关系就可以判断出函数变化的大致趋势，以作为搜索方向的参考。常见的这一类优化方法有黄金分割法（又称 0.618 优选法）、单纯形法及随机射线法等，其中单纯形法以其概念清晰、实现便利等优良性能广泛为人们所采用，下面就其优化设计原理做一简要介绍。

（1）单纯形　所谓单纯形是指变量空间内最简单的规则形体。如在二维平面内正三角形即为单纯形，而在三维空间内正四面体为单纯形。

（2）单纯形法的寻优原理　为便于说明，现以二维情况加以讨论。如图 4-14 所示，设初始单纯形 3 个顶点 A、B、C 的坐标值为（x_{1i}，x_{2i}），$i=1$，2，3，它相当于三个设计方案（例如为 PI 调节器参数的三种方案），可以分别计算出（或试验出）相应的三个目标函数值，然后比较出大小，称目标函数值最大的点为“坏点”，次之为“次坏点”，最小的点为“好点”。这样，单纯形法的寻优过程可概括为如下几点：

1）寻优规则。设 $\triangle ABC$ 的 A 点为坏点，于是抛弃 A 点，将三角形沿 BC 边翻转，得到一个新的 $\triangle A_1BC$。重复这一过程，即不断地抛弃坏点，建立新点。若坏点重复，则应抛弃次坏点，继续上述过程。

2）终点判别。当三角形逐渐接近峰顶（最优值）时，会出现三角形绕同一“好点”转圈的情况，即出现好点重复的现象。若用 T 表示单纯形绕最优点重复翻转的次数，N 表示变量的维数，则当 $T \geqslant 1.65N$ 时说明三角形已达到峰顶（极值点）。

3）精度调整。如果对所设计变量的精度要求较高，可将单纯的边长 a 值缩小，而后继续上述寻优过程，直到满意为止。

单纯形法最终所得到的“好点”即为最优设计方案，对应的变量值为最优设计参数。

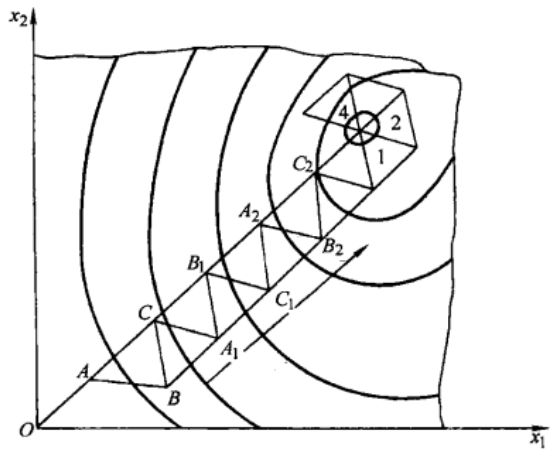


图 4-14 单纯形法寻优原理

（3）单纯形顶点的坐标计算 单纯形的翻转具有一个特点，即新得到的单纯形仅增加一个新点，其余各顶点不变。所以单纯形顶点的坐标计算关键在于求解新点的坐标，这是一个几何问题，不难推导 n 个变量时的新点坐标计算公式如下

$$\text{新点坐标} = \frac{2}{n} \times (\text{留下 } n \text{ 点坐标之和}) - (\text{抛弃点的坐标})$$

（三）目标函数的选取

对于不同的优化设计问题，其目标函数的选取方式是不一样的，下面所述目标函数的选取方式是针对图 4-15 所示控制系统而言的。

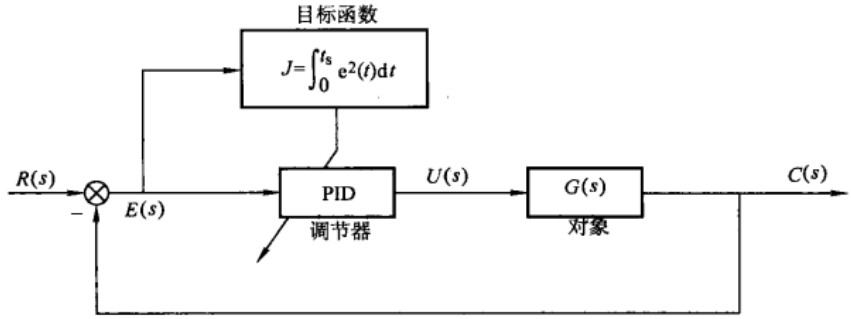


图 4-15 控制系统调节器参数优化设计原理图

控制系统参数的优化设计中，常用的目标函数有以下几种：

1）IAE 准则，即 $J = \int_0^{t_s} | e(t) | \, dt$。

2）ISE 准则，即 $J = \int_0^{t_s} e^2(t) \, dt$。

3）ITAE 准则，即 $J = \int_0^{t_s} t | e(t) | \, dt$。

4）ITSE 准则，即 $J = \int_0^{t_s} te^2(t) \, dt$。

5）ISTAE 准则，即 $J = \int_0^{t_s} t^2 | e(t) | \, dt$。

6）ISTSE 准则，即 $J = \int_0^{t_s} t^2 e^2(t)\,\mathrm{d}t$。

这些目标函数对于同一个优化问题，其优化结果是不相同的，使控制系统所具有的动态性能也不一样（如快速性、超调量等），具体应用哪一种目标函数还需在实际应用中适当地加以选择，下面举例说明之。

例 4-8　对于位置伺服系统，一般要求具有“快速-无超调”的阶跃响应特性，试选择优化设计的目标函数。

解　经验表明，ITAE 准则可使控制系统具有快速响应特性，因此所选目标函数中应含有 ITAE 准则。如何达到系统动态响应“无超调”的目的呢？我们只须在目标函数中加入对系统超调量的约束即可。则系统参数优化的目标函数为

$$J = \int_0^{t_s} t \mid e(t) \mid \mathrm{d}t + \alpha \int_0^{t_s} E(t)\,\mathrm{d}t$$

式中，$E(t) \in [(r(t) - c(t)) < 0]$；$\alpha$ 为 $E(t)$ 的加权系数。实际应用表明，只要适当选取 α 值，参数优化后系统可以达到“快速-无超调”的目的。

（四）实例分析

对基于单纯形法的无限定多变量优化问题，MATLAB 提供了可直接应用的函数，其语句格式为

X = fmins('函数名',初值),

下面通过例题说明其具体用法。

例 4-9　已知图 4-15 所示系统中，调节器为 PI 结构，对象传递函数 $G(s) = \dfrac{1}{10s+1}\mathrm{e}^{-s}$，若希望系统对单位阶跃给定具有快速响应特性，试确定调节器参数 K_p、K_i 值。

解　首先，设计一个函数，定名为 optm. m，在这个函数中以 K_p、K_i 为自变量，ITSE 准则下的目标函数为输出；采用命令行仿真方式进行计算，其仿真函数为 sim（），调用格式为

[t,x,y] = sim('. mdl 文件名',仿真时间,仿真初值)

式中，t 是仿真的时间向量；x 为状态变量矩阵；y 为 . mdl 文件的输出。

其次，使用 Simulink 建立仿真模型文件（命名为 optzhang. mdl），如图 4-16 所示。

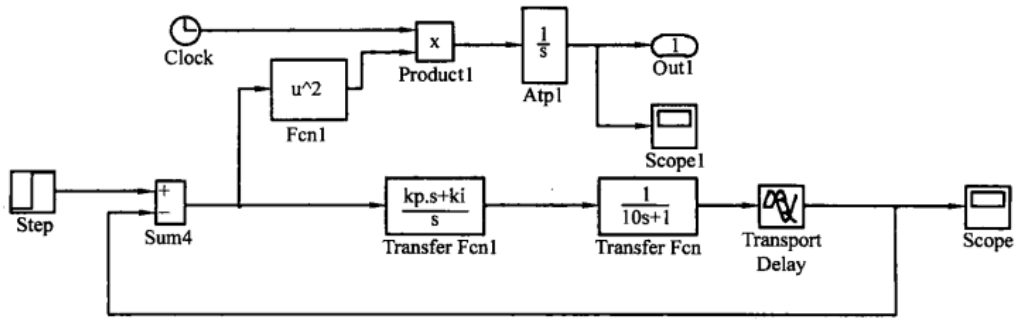


图 4-16　单纯形法优化设计的图形组态

上述设计的源程序如下：

主程序：tryopt. m

```
global kp;
global ki;
```

```
global i;
i = 1;
result = fmins ( 'optm', [2 1])
```

优化的目标函数：optm. m

```
function ss = optm (x)
global kp;
global ki;
global i;
kp = x (1);
ki = x (2);
i = i + 1
[tt, xx, yy] = sim ( 'optzhang', 40, []);
yylong = length (yy);
ss = yy (yylong);
```

优化计算时，在 MATLAB 的命令窗口运行主程序即可得到优化的参数 K_p 及 K_i，而优化后系统的动态性能可以利用 MATLAB 的各种手段进行观察与处理，图 4-17 给出了优化设计的时域仿真结果，其中曲线 1 为优化前的动态响应（$K_p=2$，$K_i=1$），曲线 2 为优化后系统的动态响应（$K_p=6.9461$，$K_i=0.58827$）。可见，系统的动态响应速度提高了很多，而且超调量也很低，系统动态性能得到有效的改善。

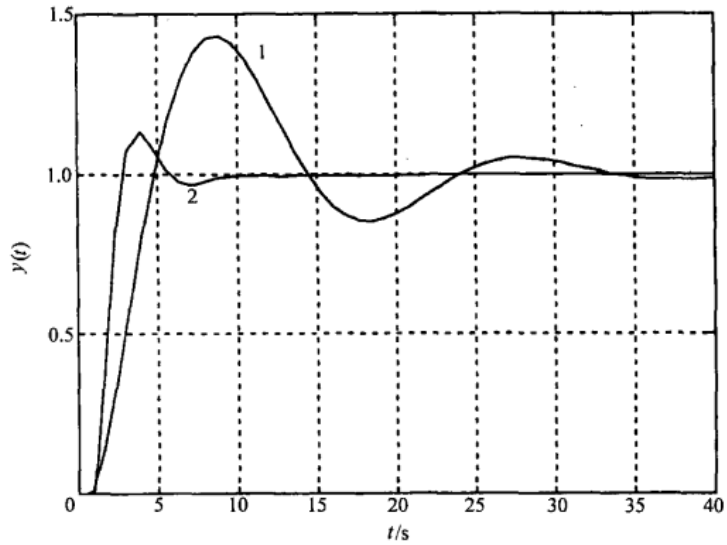


图 4-17 优化设计的时域仿真结果

（五）实际应用中的几个问题

（1）优化设计结果的有效性问题 尽管人们总是在设法使系统性能指标达到最优化，但在实际问题中还要考虑效果是否显著。比如，采用不同的目标函数进行优化设计所得到的最优参数值对控制系统固有的参数变化的敏感程度就不完全相同（如 ISE 不很灵敏，ITAE 较灵敏）。通常是将不太灵敏的指标函数应用于系统固有参数值不是太确切的优化设计中。因此，有时我们宁可让所得参数适当地偏离“最优”值，以求得其他指标的改善。

（2）局部最优与全局最优问题　在优化设计过程中应注意所得到的解可能为局部极值点，而不是我们所寻求的全局范围内的极值点。一般情况下，如果从不同的初点开始寻优，而能得到相同的结果，则称这一结果为全局最优解，否则就是局部最优解。

（3）寻优速度问题　尽管单纯形法非常适合于非线性控制系统的参数优化问题，但其寻优速度并不理想，相对较慢，在某种程度上影响了它的应用。通常人们是在初值、步长与精度三方面进行协调，以使寻优速度（或优化设计效率）得以提高，具体方法大家很容易想象出，这里就不作详谈了。

（4）“在线”应用问题　通常情况下，由于优化设计算法比较繁琐、收敛速度不理想等原因，使得优化设计仅停留在“离线”应用的水平上，即只是在计算机上作理论分析与设计。现在，由于数字计算机在硬件及软件水平上都已有较大进步，尤其是网络技术在控制系统（如分布式控制系统）中的逐步应用为优化技术的“在线”应用开辟了广阔前景。优化设计技术现已在“参数自整定温度控制器”、“位置伺服系统的自寻最优控制”及“电气控制系统运行状态的自动诊断”等问题上得到实际应用。经验表明[42]，对于一类具有重复运动特性的位置伺服系统（如定位测试系统），只要初值选择得当，基于单纯形法的调节器参数自整定方法具有良好的收敛特性，完全可以“在线”应用，以实现系统运行状态的自寻最优控制。

第三节　基于双闭环 PID 控制的一阶倒立摆控制系统设计

在图 4-18 所示的“一阶倒立摆控制系统”中，通过检测小车位置与摆杆的摆动角，来适当控制驱动电动机拖动力的大小，控制器由一台工业控制计算机（IPC）完成。

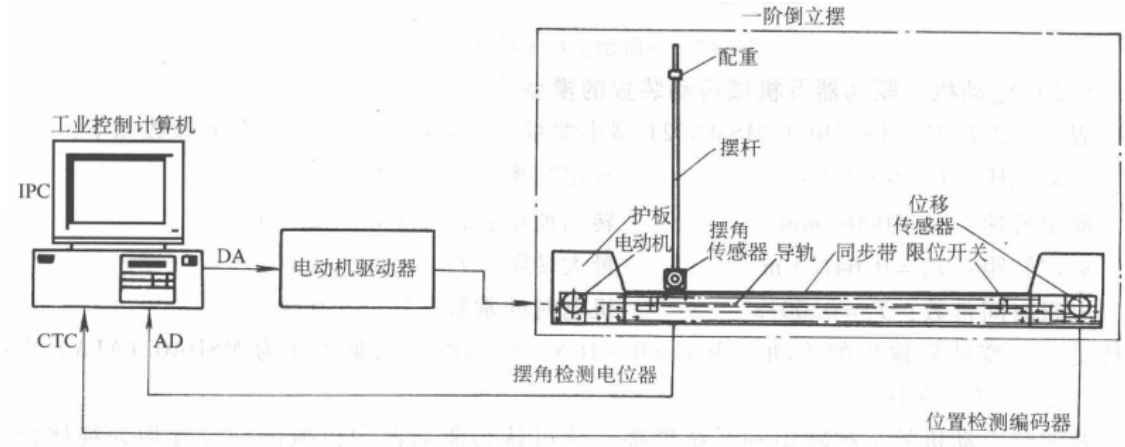


图 4-18　一阶倒立摆控制系统

本节将借助于“Simulink 封装技术——子系统”，在模型验证的基础上，采用双闭环 PID 控制方案，实现倒立摆位置伺服控制的数字仿真实验。

一、系统模型

（一）对象模型

由第二章第二节可知：

1）一阶倒立摆精确模型为

$$\begin{cases} \ddot{x} = \dfrac{(J+ml^2)F + lm(J+ml^2)\sin\theta\cdot\dot{\theta}^2 - m^2l^2g\sin\theta\cos\theta}{(J+ml^2)(m_0+m) - m^2l^2\cos^2\theta} \\ \ddot{\theta} = \dfrac{ml\cos\theta\cdot F + m^2l^2\sin\theta\cos\theta\cdot\dot{\theta}^2 - (m_0+m)mlg\sin\theta}{m^2l^2\cos^2\theta - (m_0+m)(J+ml^2)} \end{cases}$$

当小车的质量 $m_0=1\text{kg}$；倒摆振子的质量 $m=1\text{kg}$；倒摆长度 $2l=0.6\text{m}$；重力加速度取 $g=10\text{m/s}^2$ 时得

$$\begin{cases} \ddot{x} = \dfrac{0.12F + 0.036\sin\theta\cdot\dot{\theta}^2 - 0.9\sin\theta\cos\theta}{0.24 - 0.09\cos^2\theta} \\ \ddot{\theta} = \dfrac{0.3\cos\theta\cdot F + 0.09\sin\theta\cos\theta\cdot\dot{\theta}^2 - 6\sin\theta}{0.09\cos^2\theta - 0.24} \end{cases}$$

2）若只考虑 θ 在其工作点 $\theta_0=0$ 附近（$-10°<\theta<10°$）的细微变化，这时可近似认为

$$\begin{cases} \dot{\theta}^2 \approx 0 \\ \sin\theta \approx \theta \\ \cos\theta \approx 1 \end{cases}$$

由此得到简化的近似模型为

$$\begin{cases} \ddot{x} = -6\theta + 0.8F \\ \ddot{\theta} = 40\theta - 2.0F \end{cases}$$

其等效动态结构图如图 4-19 所示。

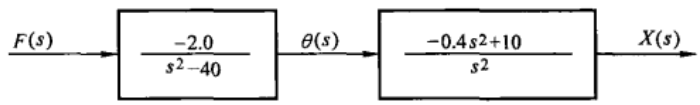


图 4-19 一阶倒立摆系统动态结构图

（二）电动机、驱动器及机械传动装置的模型

假设：选用日本松下电工 MSMA021 型小惯量交流伺服电动机，其有关参数如下：

驱动电压：$U=0\sim100\text{V}$　　额定功率：$P_N=200\text{W}$

额定转速：$n=3000\text{r/min}$　　转动惯量：$J=3\times10^{-6}\text{kg}\cdot\text{m}^2$

额定转距：$T_N=0.64\text{N}\cdot\text{m}$　　最大转距：$T_M=1.91\text{N}\cdot\text{m}$

电磁时间常数：$T_l=0.001\text{s}$　　机电时间常数：$T_m=0.003\text{s}$

经传动机构变速后输出的拖动力为 $F=0\sim16\text{N}$；与其配套的驱动器为 MSDA021A1A，控制电压 $U_{DA}=(0\sim\pm10)\text{V}$。

若忽略电动机的空载转矩和系统摩擦，就可认为驱动器和机械传动装置均为纯比例环节，并假设这两个环节的增益分别为 K_d 和 K_m。

对于交流伺服电动机，其传递函数可近似为

$$\frac{K_v}{T_mT_ls^2 + T_ms + 1}$$

由于是小惯性的电动机，其时间常数 T_l、T_m 相对都很小，这样可以进一步将电动机模型近似等效为一个比例环节 K_v。

综上所述，电动机、驱动器、机械传动装置三个环节就可以合成为一个比例环节

$$G(s) = K_d K_v K_m = K_s$$

$$K_s = F_{max}/U_{max} = 16/10 = 1.6$$

二、模型验证

尽管上述数学模型系经机理建模得出，但其准确性（或正确性）还需运用一定的理论与方法加以验证，以保证以其为基础的仿真实验的有效性。模型验证的理论与方法是一专门技术，本书受篇幅所限不能深入阐述。下面给出的是一种“必要条件法”，即我们所进行的模型验证实验的结果是依据经验可以判定的，其正确的结果是“正确的模型”所应具备的“必要性质”。

（一）Simulink 子系统

如同大多数程序设计语言中的子程序（如 C 语言中的函数、MATLAB 中的 M 文件）功能，Simulink 中也有类似的功能——子系统。子系统通过将大的复杂的模型分割成几个小的模型系统，使得整个系统模型更加简捷，可读性更高。

把已存在的 Simulink 模型中的某个部分或全部封装成子系统的操作程序如下：

1）首先使用范围框将要“封装”成子系统的部分选中，包括模块和信号线（见图 4-20）。为了使范围框圈住所需要的模块，常常需要事先重新安排各模块的位置（注意：这里只能用范围框，而不能用 Shift 键逐个选定）。

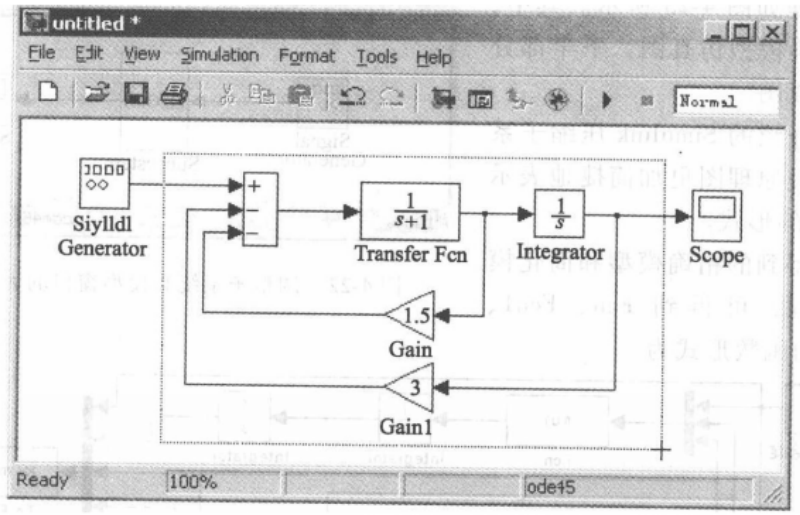


图 4-20　选择要封装的模块

2）在模块窗口菜单选项中选择［Edit >> Create Subsystem］，Simulink 将会用一个子系统模块代替选中的模块组，如图 4-21 所示。

3）所得子系统模块将有默认的输入和输出端口。输入端口和输出端口的默认名称分别为 In1 和 Out1。调整子系统和模型窗口的大小，使之更加美观，如图 4-22 所示。

若想查看子系统的内容或对子系统进行再编辑，可以双击子系统模块，则会出现一个显示子系统内容的新窗口。在窗口内除了原始模块外，Simulink 自动添加输入模块和输出模块，分别代表子系统的输入端口和输出端口。改变其标签会使子系统的输入输出端口的标签也随之变化。

这里需要注意的是菜单命令［Edit >> Create Subsystem］没有相反的操作命令，即一旦

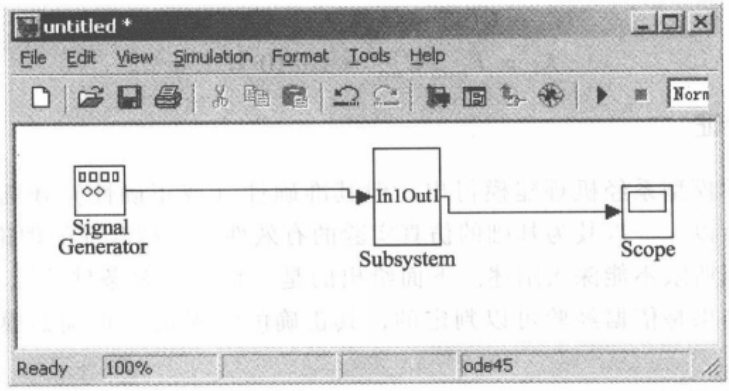


图 4-21 封装后的模型图

将一组模块封装成子系统，就没有可以直接还原的处理方法了（UNDO 除外）。因此，在封装子系统前应将模型保存，作为备份。

（二）仿真验证

1. 模型封装

采用仿真实验的方法在 MATLAB 的 Simulink 图形仿真环境下进行模型验证实验，其原理如图 4-23 所示。其中，上半部分为精确模型仿真图，下半部分为简化模型仿真图。

利用前面介绍的 Simulink 压缩子系统功能可将验证原理图更加简捷地表示为图 4-24 所示的形式。

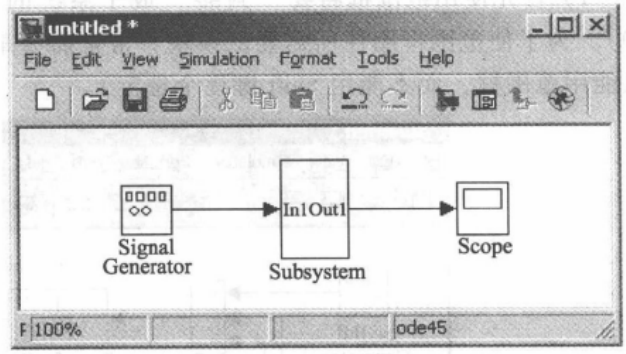


图 4-22 调整子系统和模型窗口的大小和方位

其中，由得到的精确模型和简化模型的状态方程，可得到 Fcn、Fcn1、Fcn2 和 Fcn3 的函数形式为

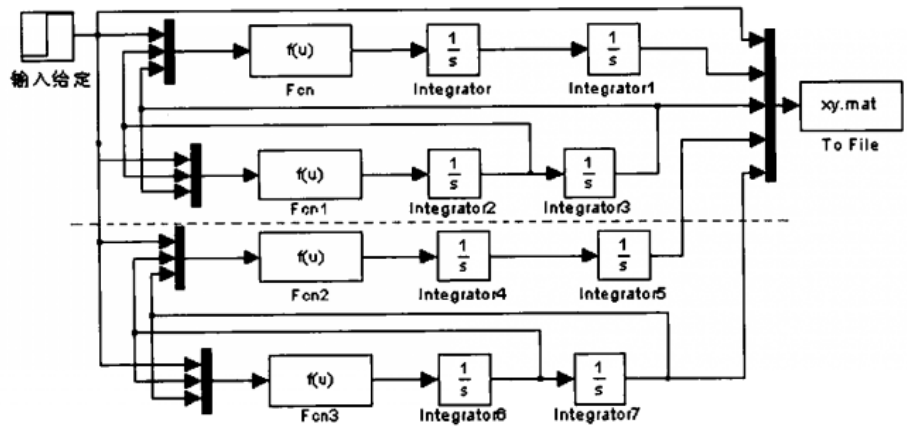


图 4-23 模型验证原理图

Fcn:(0.12 * u[1] + 0.036 * sin(u[3]) * power(u[2],2) - 0.9 * sin(u[3]) * cos(u[3]))/(0.24 - 0.09 * power(cos(u[3]),2))

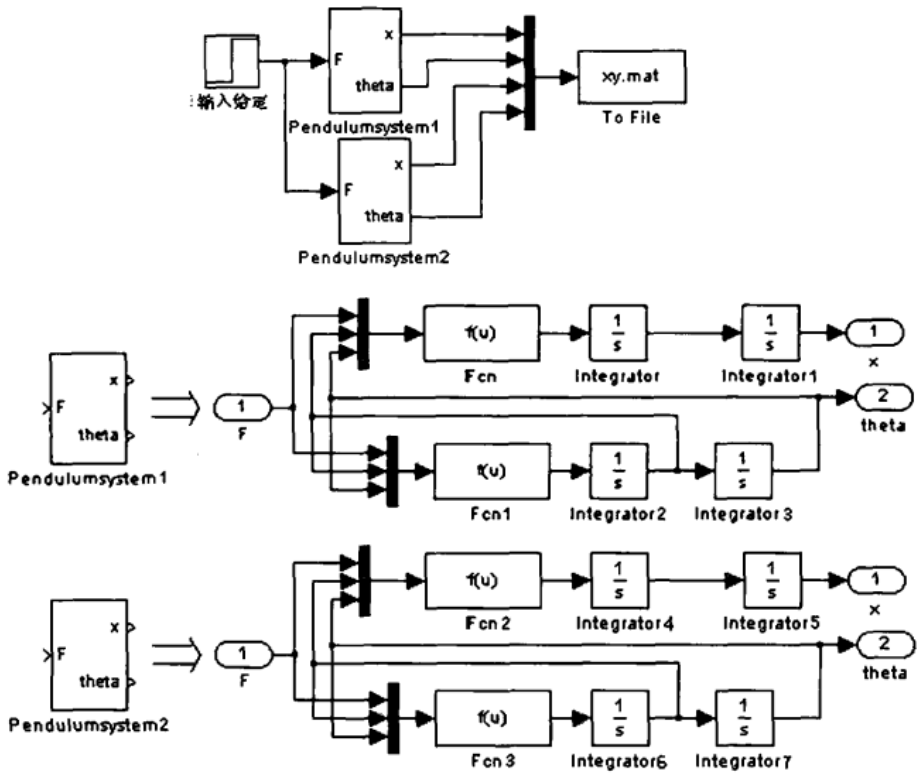


图 4-24　利用子系统封装后的框图

Fcn1:(0.3 * cos(u[3]) * u[1] +0.09 * sin(u[3]) * cos(u[3]) * power(u[2],2) -6 * sin(u[3]))/(0.09 * power(cos(u[3]),2) -0.24)

Fcn2:0.8 * u[1] -6 * u[3]

Fcn3:40 * u[3] -2.0 * u[1]

2. 实验设计

假定使倒立摆在($\theta=0, x=0$)初始状态下突加微小冲击力作用,则依据经验知,小车将向前移动,摆杆将倒下。下面利用仿真实验来验证正确数学模型的这一"必要性质"。

3. 编制绘图子程序

```
% Inverted pendulum
% Model tesl in open loop
% Singnals recuperation
%将导入到 xy. mat 中的仿真试验数据读出
load xy. mat
t = signals(1,:);                    %读取时间信号
f = signals(2,:);                    %读取作用力 F 信号
x = signals(3,:);                    %读取精确模型中的小车位置信号
q = signals(4,:);                    %读取精确模型中的倒立摆摆角信号
xx = signals(5,:);                   %读取简化模型中的小车位置信号
qq = signals(6,:);                   %读取简化模型中的倒立摆摆角信号
```

```
% Drawing control and x (t) response signals
% 画出在控制力的作用下的系统响应曲线
% 定义曲线的横纵坐标、标题、坐标范围和曲线的颜色等特征
figure (1)                                          % 定义第一个图形
hf = line (t, f (:));                               % 连接时间-作用力曲线
grid on;
xlabel ('Time (s)')                                 % 定义横坐标
ylabel ('Force (N)')                                % 定义纵坐标
axis ([0 1 0 0.12])                                 % 定义坐标范围
axet = axes ('Position', get (gca,'Position'), ...
             'XAxisLocation','bottom', ...
             'YAxisLocation','right','Color','None', ...
             'XColor','k','YColor','k');            % 定义曲线属性
ht = line (t, x,'color','r','parent', axet);        % 连接时间-小车位置曲线
ht = line (t, xx,'color','r','parent', axet);       % 连接时间-小车速度曲线
ylabel ('Evolution of the x position (m)')           % 定义坐标名称
axis ([0 1 0 0.1])                                  % 定义坐标范围
title ( 'Response x and x'' in meter to a f (t) pulse of 0.1 N')  % 定义曲线标题名称
gtext ('\ leftarrow f (t)'), gtext ('x (t) \ rightarrow') , gtext ('\ leftarrow x'' (t)')
% drawing control and theta (t) response singals
figure (2)
hf = line (t, f (:));
grid on
xlabel ('Time')
ylabel ('Force in N')
axis ([0 1 0 0.12])
axet = axes ('Position', get (gca,'Position'), ...
             'XAxisLocation','bottom', ...
             'YAxisLocation','right','Color','None', ...
             'XColor','k','YColor','k');
ht = line (t, q,'color','r','parent', axet);
ht = line (t, qq,'color','r','parent', axet);
ylabel ('Angle evolution (red)')
axis ([0 1 -0.3 0])
title('Response \theta(t) and \theta''(t) in rad to a f(t)pulse of 0.1 N')
gtext('\leftarrow f(t)'),gtext('\theta(t)\rightarrow'),gtext('\leftarrow \theta''(t)')
```

4. 仿真实验

执行该程序的结果如图 4-25 所示。从中可见，在 0.1N 的冲击力作用下，摆杆倒下（θ 由零逐步增大），小车位置逐渐增加，这一结果符合前述的实验设计，故可以在一定程度上

确认该“一阶倒立摆系统”的数学模型是有效的。同时，由图中也可看出，近似模型在0.8s以前与精确模型非常接近，因此，也可以认为近似模型在一定条件下可以表述原系统模型的性质。

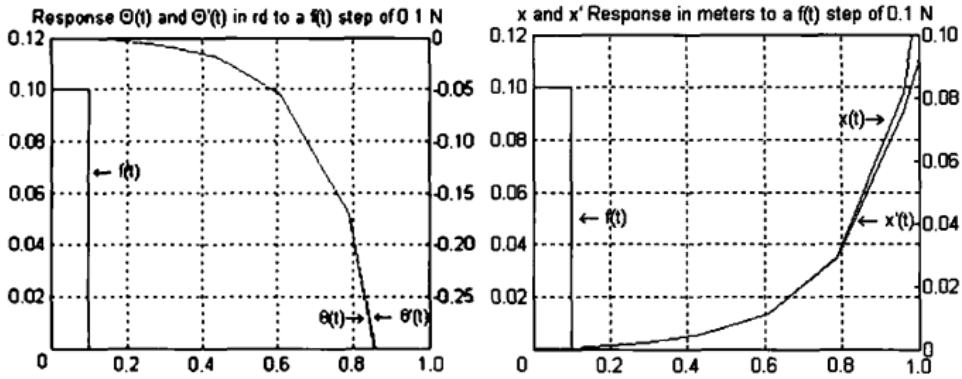


图 4-25　模型验证仿真结果

三、双闭环 PID 控制器设计

从图 4-19 所示的一阶倒立摆系统动态结构图中不难看出，对象传递函数中含有不稳定的零极点，即该系统为一个“自不稳定的非最小相位系统”。

由于一阶倒立摆系统位置伺服控制的核心是“在保证摆杆不倒的条件下，使小车位置可控”（注：此处本应证明系统的“可控性”，受篇幅所限，感兴趣的读者请见本章第四节）。因此，依据负反馈闭环控制原理，将系统小车位置作为“外环”，而将摆杆摆角作为“内环”，则摆角作为外环内的一个扰动，能够得到闭环系统的有效抑制（实现其直立不倒的自动控制）。

综上所述，设计一阶倒立摆位置伺服控制系统如图 4-26 所示。剩下的问题就是如何确定控制器（校正装置）$D_1(s)/D_1'(s)$ 和 $D_2(s)/D_2'(s)$ 的结构与参数。

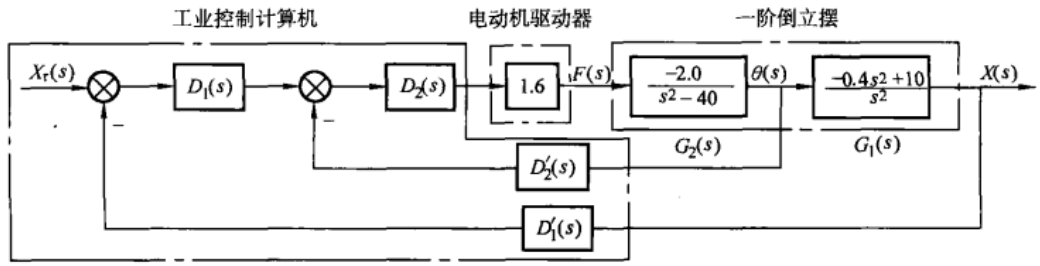


图 4-26　一阶倒立摆位置伺服控制系统动态结构图

（一）内环控制器的设计

1. 控制器结构的选择

考虑到对象为一个非线性的自不稳定系统，故拟采用反馈校正，这是因为其具有如下特点：

1）削弱系统中非线性特性等不希望特性的影响。

2）降低系统对参数变化的敏感性。

3）抑制扰动。

4）减小系统的时间常数。

所以，我们对系统内环采用反馈校正进行控制。

图 4-27 为采用反馈校正控制的系统内环框图。其中，K_s 为伺服电动机与减速机构的等效模型（已知 $K_s=1.6$），反馈控制器 $D_2'(s)$ 可有 PD、PI、PID 三种形式，那么应该采用什么形式的反馈校正装置（控制器）呢？下面，我们采用绘制各种控制器下的闭环系统根轨迹的方法进行分析，以选出一种适合的控制器结构。

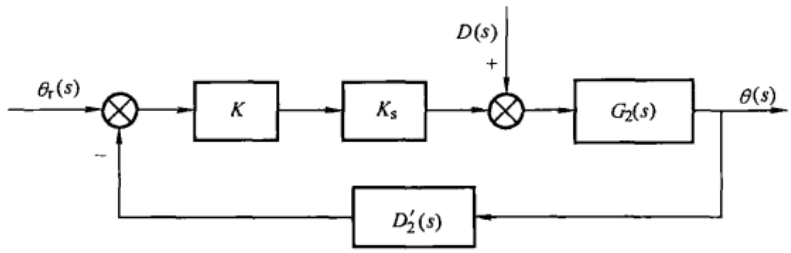
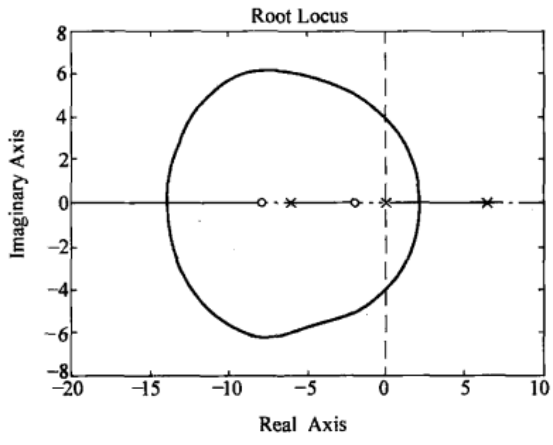
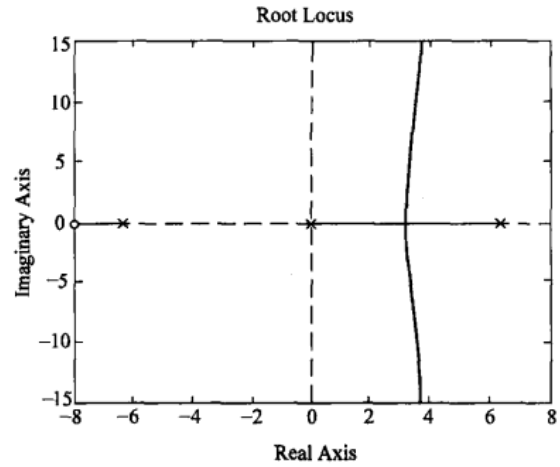


图 4-27 反馈校正控制的系统内环框图

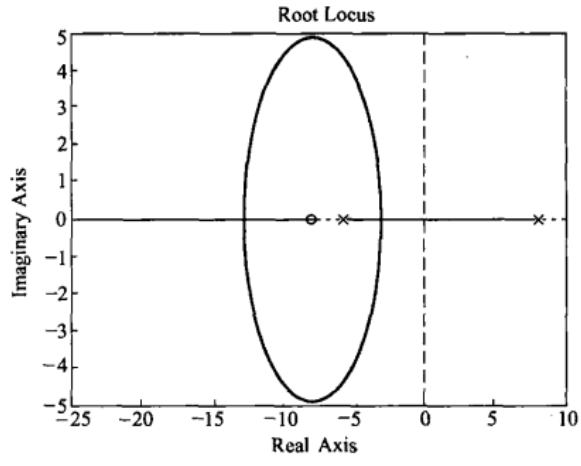
图 4-28 给出了各种控制器结构下内环系统的根轨迹，其中暂定 $D_2(s)=K$ 为纯比例环



a)



b)



c)

图 4-28 各种控制器下的内环根轨迹

a）PID 控制 b）PI 控制 c）PD 控制

节。从图中不难看出，采用 PD 结构的反馈控制器可使系统结构简单，使原先不稳定的系统稳定，所以，我们选定反馈校正装置的结构为 PD 结构的控制器。

综上有 $D_2'(s)=K_{p2}+K_{d2}s$，同时为了加强对干扰量 $D(s)$ 的抑制能力，我们在前向通道上加一个比例环节 $D_2(s)=K$，从而有系统内环动态结构如图 4-29 所示。

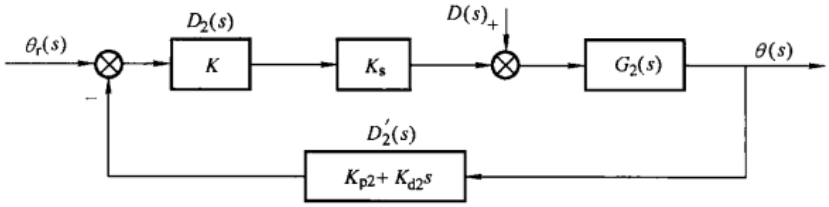


图 4-29　系统内环动态结构框图

2. 控制器参数的整定

首先暂定比例环节 $D_2(s)$ 的增益 $K=-20$，又已知 $K_s=1.6$。这样可以求出内环的传递函数为

$$W_2=\frac{KK_sG_2(s)}{1+KK_sG_2(s)D_2'(s)}$$

$$=\frac{-20\times1.6\times\dfrac{-2.0}{s^2-40}}{1+(-20)\times1.6\times\dfrac{-2.0}{s^2-40}(K_{p2}+K_{d2}s)}$$

$$=\frac{64}{s^2+64K_{d2}s+64K_{p2}-40}$$

由于对系统内环的特性并无特殊的指标要求，因此对于这一典型的二阶系统，采取典型参数整定办法，即以保证内环系统具有“快速跟随性能特性”（使阻尼比 $\zeta=0.7$，闭环增益 $K=1$ 即可）为条件来确定反馈控制器的参数 K_{p2} 和 K_{d2}，这样就有

$$\begin{cases}64K_{p2}-40=64\\64K_{d2}=2\times0.7\times\sqrt{64}\end{cases}$$

由上式得

$$\begin{cases}K_{p2}=1.625\\K_{d2}=0.175\end{cases}$$

系统内环的闭环传递函数为

$$W_2(s)=\frac{64}{s^2+11.2s+64}$$

3. 系统内环的动态跟随性能指标

（1）理论分析　系统内环的动态跟随性能指标如下：

固有频率 $\omega_n=\sqrt{64}\text{rad/s}=8\text{rad/s}$

阻尼比 $\zeta=0.7$

超调量 $\sigma\%=e^{\frac{-\zeta\pi}{\sqrt{1-\zeta^2}}}\times100\%=4.6\%$

调节时间 $t_s \approx \frac{3}{\zeta\omega_n} = 0.536\text{s}$（5% 允许误差所对应的 t_s）

（2）仿真实验　根据得到的内环系统的闭环传递函数，很容易搭建 Simulink 仿真模型，如图 4-30 所示。

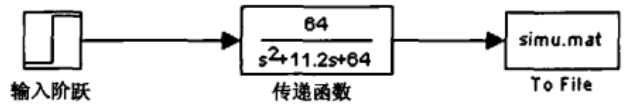


图 4-30　搭建的 Simulink 仿真图

编写绘图子程序如下：

```
% 将导入到 simu. mat 中的仿真试验数据读出
load simu. mat
t = signals (1,:);
x = signals (2,:);
hf = line (t, x (:));
figure (1)
axis([0 2 0 1.2])
grid on
xlabel ('Time (s)')
ylabel ('The response of the step signal (100%)')
title ('Respones')
```

得到的仿真图形如图 4-31 所示。

从仿真图中可以很清楚地得知，其响应时间和超调量与理论分析的值相符合。

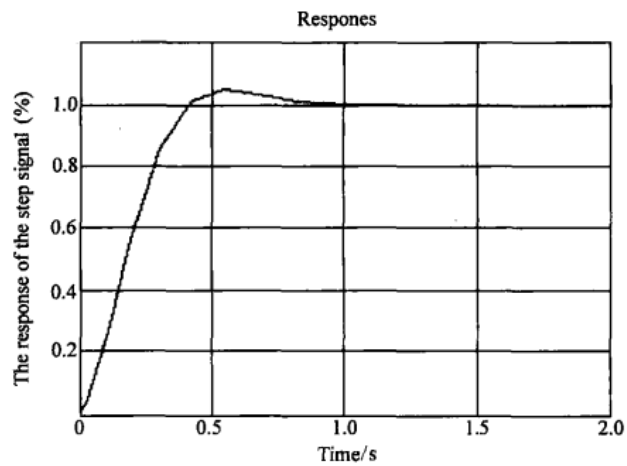


图 4-31　单位阶跃信号作用下的响应曲线

（二）系统外环控制器设计

外环系统前向通道的传递函数为

$$W_2(s)G_1(s) = \frac{64}{s^2+11.2s+64} \times \frac{-0.4s^2+10}{s^2} = \frac{64(-0.4s^2+10)}{s^2(s^2+11.2s+64)}$$

可见，系统开环传递函数可视为一个高阶（四阶）且带有不稳定零点的“非最小相位系统”，为了便于设计，需要先对它进行一些必要的简化处理（否则，不便利用经典控制理论与方法对其进行设计）。

1. 系统外环模型的降阶[6]

对于一个高阶系统，当高次项的系数小到一定程度时，该环节对系统的影响可忽略不计。这样可降低系统的阶次，以使系统得到简化。

（1）对内环等效闭环传递函数的近似处理　由上可知，系统内环闭环传递函数为

$$W_2(s) = \frac{64}{s^2 + 11.2s + 64}$$

若可以将高次项 s^2 忽略，则可以得到近似的一阶传递函数为

$$W_2(s) \approx \frac{64}{11.2s + 64} = \frac{1}{0.175s + 1}$$

近似条件可以由频率特性导出，即

$$W(j\omega) = \frac{64}{(j\omega)^2 + 11.2(j\omega) + 64}$$

$$= \frac{64}{(64 - \omega^2) + 11.2j\omega} \approx \frac{64}{64 + 11.2j\omega}$$

所以，近似条件是 $\omega_c^2 \leqslant \frac{64}{10}$，即 $\omega_c \leqslant 2.52$。

（2）对象模型 $G_1(s)$ 的近似处理　我们知道对于 $G_1(s) = \frac{-0.4s^2 + 10}{s^2}$，如果可以将分子中的高次项（$-0.4s^2$）忽略，则环节可近似为二阶环节，即 $G_1(s) \approx 10/s^2$。

同理，近似条件是 $0.4\omega_c^2 \leqslant \frac{10}{10}$，即 $\omega_c \leqslant 1.58$。

经过以上的处理后，系统开环传递函数被简化为

$$W_2(s)\ G_1(s) \approx \frac{57}{s^2(s + 5.7)}$$

近似条件为 $\omega_c \leqslant \min(2.52, 1.58) = 1.58$。

2. 控制器设计[6]

图 4-32 给出了系统外环前向通道上传递函数的等效过程，从最终的简化模型上不难看出，这是一个二型系统。鉴于一阶倒立摆位置伺服控制系统对抗扰性能与跟随性能的要求（对摆杆长度、质量的变化应具有一定的抑制能力，同时可使小车有效定位），可以将外环系统设计成典型Ⅱ型的结构形式。同时，系统还应满足前面各环节的近似条件，即系统外环的截止角频率 $\omega_c \leqslant 1.58$。

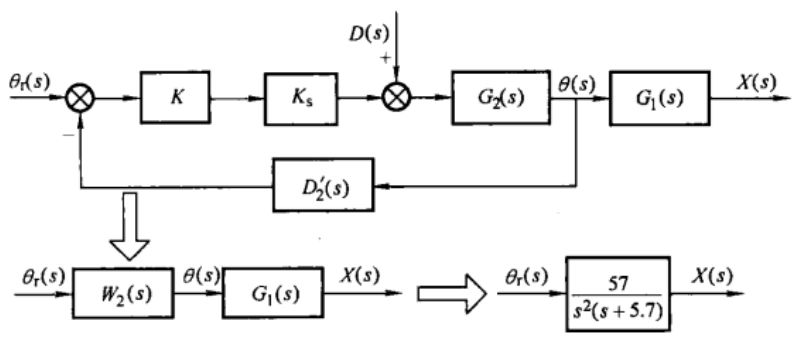


图 4-32　模型简化过程

为了满足以上对系统的设计要求，不难发现所需要加入的调节器 $D_1(s)$ 也应为 PD 的形式。设加入的调节器为 $D_1(s) = K_p(\tau s + 1)$，同时，为使系统有较好的跟随性能，采用单位

反馈($D_1'(s)=K=1$)来构成外环反馈通道，如图 4-33 所示。此时系统的开环传递函数为

$$W(s)=W_2(s)G_1(s)D_1(s)=\frac{57}{s^2(s+5.7)}K_p(\tau s+1)$$

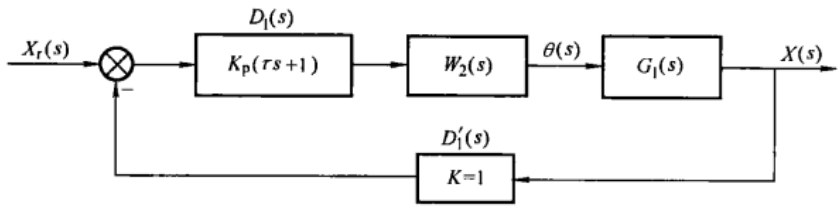


图 4-33 闭环系统结构图

为保证系统剪切频率 $\omega_c \leqslant 1.58$，我们不妨取 $\omega_c=1.2$，则由“典型Ⅱ型”系统最佳参数（中频宽 $h=5$）知：

$h=\frac{\tau}{1/5.7}$，则 $\tau=\frac{5}{5.7}=0.877$，取 $\tau=1$，则有系统开环传递函数 $W(s)=\frac{10K_p(s+1)}{s^2(0.175s+1)}$；再由“典型Ⅱ型”系统伯德图特性（$K=\omega_1\omega_c$）知：$10K_p=1\times1.2$，则有 $K_p=0.12$。

综上所述，我们求出外环调节器的两个参数：$K_p=0.12$，$\tau=1$，这样我们得到完整的系统仿真结构如图 4-34 所示。

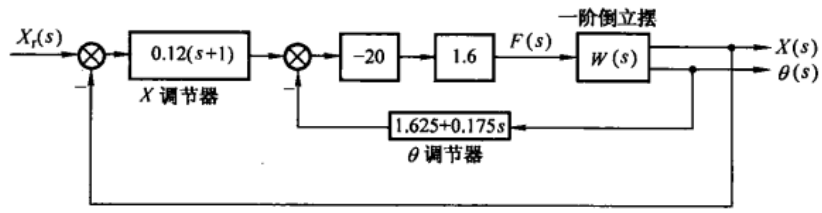


图 4-34 系统仿真结构图

四、仿真实验

综合上述内容，有图 4-35 所示的 Simulink 仿真系统结构图。需要强调的是，其中的对象模型为精确模型的封装子系统形式。

系统仿真绘图子程序及仿真结果如下：

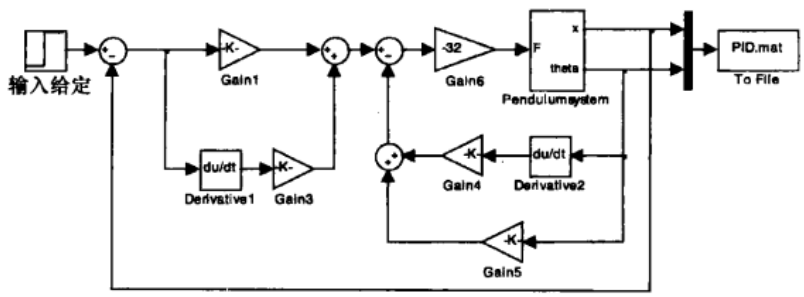


图 4-35 Simulink 仿真框图

（一）绘图子程序

```
% Inverted PID
% singnals recuperation
%将导入到 PID. mat 中的仿真实验数据读出
load PID. mat
t = signals (1,:);
q = signals (2,:);
x = signals (3,:);
% drawing x (t) and theta (t) response signals
%画小车位置和摆杆角度的响应曲线
figure (1)
hf = line (t, q (:));
grid on
xlabel ('Time (s)')
ylabel ('Angle evolution (rad)')
axis ( [0 10 -0.3 1.2])
axet = axes ('Position', get (gca,'Position'), ...
             'XAxisLocation','bottom', ...
             'YAxisLocation','right','Color','None', ...
             'XColor','k','YColor','k');
ht = line (t, x,'color','r','parent', axet);
ylabel ('Evolution of the x position (m)')
axis ( [0 10 -0.3 1.2])
title ('\ theta (t) and x (t) Response to a step input')
gtext ('\ leftarrow x (t)'), gtext ('\ theta (t) \ uparrow')
```

（二）仿真结果

图 4-36 给出了仿真实验结果，从中可见，双闭环 PID 控制方案是有效的。

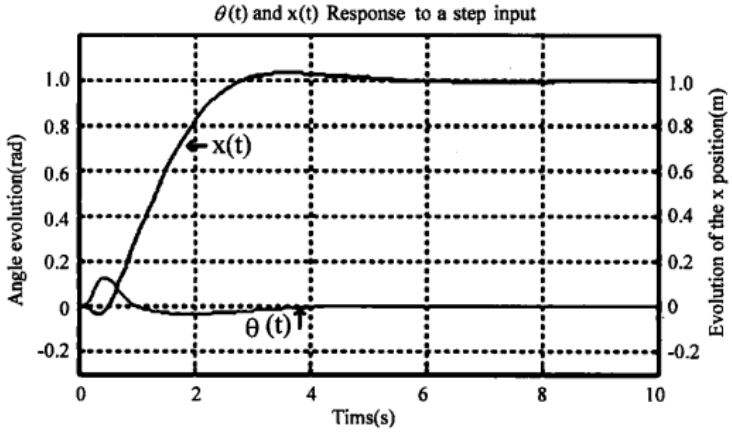


图 4-36　系统仿真结果图

为检验控制系统的鲁棒性能，还可以改变一下倒立摆系统的部分参数来检验一下系统是否具有一定的鲁棒性。例如，将倒立摆的摆杆质量改为 1.1kg，此时的 Simulink 仿真框图仍如图 4-35 所示，只是将 Fcn 和 Fcn1 作如下修改：

Fcn：(0.132 * u [1] +0.0436 * sin (u [3]) * power (u [2], 2) -1.090 * sin (u [3]) * cos (u [3])) / (0.2772 -0.1090 * power (cos (u [3]), 2))

Fcn1：(0.33 * cos (u [3]) * u [1] +0.109 * sin (u [3]) * cos (u [3]) * power (u [2], 2) -6.93 * sin (u [3])) / (0.109 * power (cos (u [3]), 2) -0.2772)

其仿真结果如图 4-37 所示。

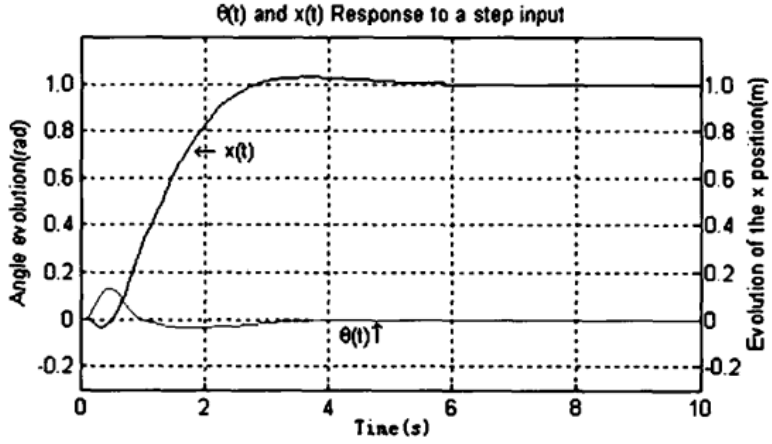


图 4-37 变参数时系统的仿真结果

从仿真结果可见，控制系统仍能有效地控制其保持倒立摆直立，并使小车移动到指定位置，系统控制是有效的。

为了进一步验证控制系统的鲁棒性，并便于进行比较，不妨改变倒立摆的摆杆质量和长度多做几组实验。部分仿真实验的结果如图 4-38 和图 4-39 所示。

可见，所设计的双闭环 PID 控制器在系统参数的一定变化范围内能有效地工作，保持摆杆直立，并使小车有效定位，控制系统具有一定的鲁棒性。

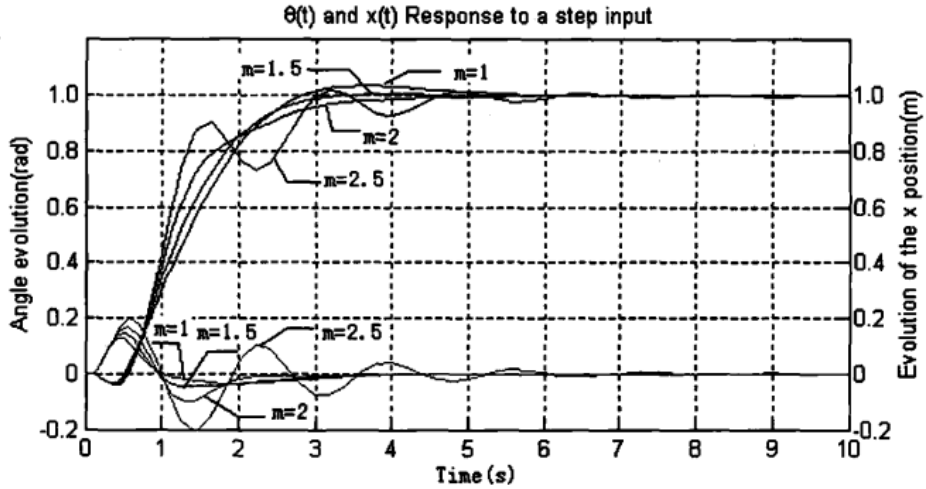


图 4-38 摆杆长度不变而摆杆质量变化时系统仿真结果

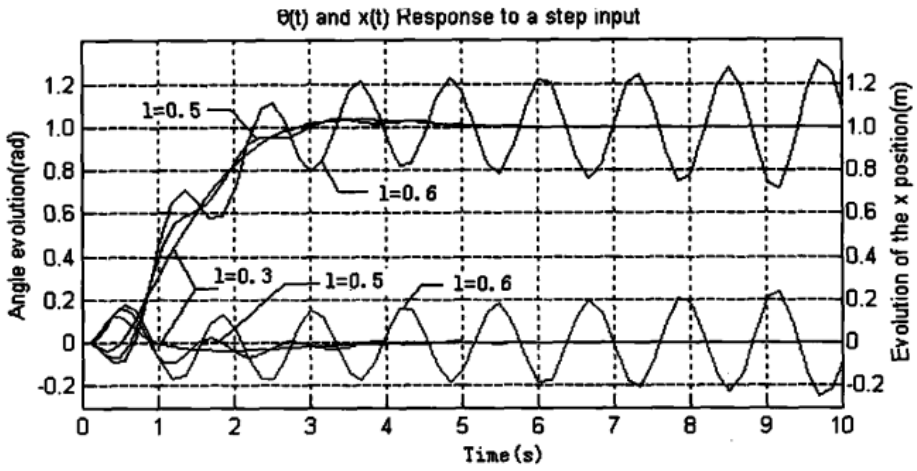


图 4-39　摆杆质量不变而摆杆长度变化时系统的仿真结果

五、结论

1）本节从理论上证明了所设计的“一阶直线倒立摆”双闭环 PID 控制方案是可行的。

2）本节的结果在实际应用时（实物仿真）还有如下问题：

① 微分控制规律易受“噪声”干扰，具体实现时应充分考虑信号的数据处理问题。

② 如采用模拟式旋转电位器进行摆角检测，在实际应用中检测精度不佳。

③ 实际应用中还需考虑初始状态下的起摆过程控制问题。

3）一阶直线倒立摆的控制问题是一个非常典型且具有明确物理意义的运动控制系统问题，对其深入的分析与应用研究，有助于提高我们分析问题与解决问题的能力。

第四节　现代控制理论 CAD

我们把以线性代数为数学工具，用状态空间法描述系统内部的动力学性能，研究系统的稳定性、能控性、能观性等定性问题，用极点配置、最优控制、状态反馈等理论及方法设计与分析控制系统概括为现代控制理论。

现代控制理论是在经典控制理论的基础上发展起来的，但又不同于经典控制理论，主要表现在以下几个方面：

1）现代控制理论是以多变量、线性及非线性系统为研究对象的。在近代的工业过程控制、飞行器控制等许多领域中，被控对象变得日趋复杂，其中包括了多变量耦合问题、参数时变问题和非线性问题。现代控制理论正是为了处理这样一类复杂的控制系统而发展起来的。

2）现代控制理论是以时域中的状态空间方法对系统进行数学描述的，并在此基础上对系统进行各种定性的和定量的分析以及希望的控制规律设计。

3）现代控制理论以现代数学方法为主要分析手段，如线性代数、微分方程及微分几何等现代数学理论在最优控制、离散事件系统、非线性系统的控制问题中都有广泛的应用，甚至诸如模糊、混沌及神经元等最新的数学方法也已经在许多控制系统分析与设计问题中得到应用并取得良好的效果。

4）现代控制理论的研究与应用是以计算机为主要工具的，计算机技术的迅猛发展极大地促进了其广泛深入的应用与推广。

5）现代控制理论是以 Pontriagin 极大值原理、Bellman 动态规划以及 Kalman 滤波技术为其形成的标志。

现代控制理论的内容较多，而 MATLAB（及其工具箱）中几乎包含了所有控制理论与控制系统分析、设计的内容，这为现代控制理论 CAD 带来极大的便利。因此，本节将以“线性二次型最优控制”和“模型参考自适应控制”为主要内容，进一步说明 MATLAB（控制系统工具箱）与 Simulink 的应用方法，以使我们能够达到触类旁通的目的。

一、线性二次型最优控制器设计

线性二次型最优控制器设计方法是 20 世纪 60 年代发展起来的一种应用较多的最优控制系统设计方法。设对象是以状态空间形式给出的线性系统，而目标函数为对象状态和控制输入的二次型函数。二次型问题就是在线性系统的约束条件下，选择适当的控制输入使得二次型目标函数达到最小。

各种问题的二次型公式的优点是它们可以导出易于实现和分析的线性控制率。我们现主要考虑二次型调节器的问题。假定系统在平衡点处，并期望在有干扰的情况下，也能维持在平衡点（即设定点）。这样，控制的目标就是使外部干扰对系统的影响最小，这也可以与跟踪或伺服控制问题相比较，它们的目标是跟踪给定参考输入或其他外部输入。跟踪问题也可以转换为二次型调节器问题。

（一）基本原理

设线性定常系统状态方程为

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) \tag{4-1}$$

式中 $\boldsymbol{x}(t)$为 n 维状态向量；$\boldsymbol{u}(t)$为 r 维控制向量；$\boldsymbol{A}$ 为 $n \times n$ 型常量矩阵；$\boldsymbol{B}$ 为 $n \times r$ 型常量矩阵，并假定控制向量 $\boldsymbol{u}(t)$ 不受约束。

二次型性能指标为

$$J = \int_{t_0}^{t_f} L(\boldsymbol{x},\boldsymbol{u})\,\mathrm{d}t$$

式中，$L(\boldsymbol{x},\boldsymbol{u})$为 $\boldsymbol{x}$ 和 $\boldsymbol{u}$ 的二次型函数。若终端时间 t_f 趋于∞，则系统属于无限长时间状态调节器问题，可以证明，由此可导出线性控制律为

$$\boldsymbol{u}(t) = -\boldsymbol{K}\boldsymbol{x}(t) \tag{4-2}$$

式中，$\boldsymbol{K}$ 为 $r \times n$ 型矩阵。

因此，基于这种二次型性能指标的最优控制系统的设计，就简化为矩阵 $\boldsymbol{K}$ 中元素的求取。

具体二次型性能指标如下

$$J = \int_0^{\infty} (\boldsymbol{x}^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{u}^{\mathrm{T}}\boldsymbol{R}\boldsymbol{u})\,\mathrm{d}t \tag{4-3}$$

式中，$\boldsymbol{Q}$ 为 $n \times n$ 型半正定实对称常数矩阵；$\boldsymbol{R}$ 为 $r \times r$ 型正定实对称常数矩阵。最优控制的目标就是求取 $\boldsymbol{u}$（t），使得性能指标式（4-3）达到最小值。

求解此类问题有许多种方法。这里采用的是基于 Lyapunov 第二方法的求解方法。

通常，先进行控制系统设计，然后再检查其稳定性。但是，也可以先给出稳定性条件，然后再在这些限制条件下设计系统。如果将 Lyapunov 第二方法作为最优控制器设计基础的话，则可以保证系统是渐进稳定的。

对于一大类控制系统来说，已经证明在 Lyapunov 函数与用于最优控制设计的二次型性能指标之间具有一种直接的关系。

式（4-3）中的矩阵 $\boldsymbol{Q}$ 和 $\boldsymbol{R}$ 决定了系统误差与控制能量消耗之间的相对重要性。下面来求解此最优控制问题。将式（4-2）代入式（4-1），可得

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} - \boldsymbol{B}\boldsymbol{K}\boldsymbol{x} = (\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K})\boldsymbol{x}$$

在下面的推导中，设矩阵 $\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}$ 是稳定的，即 $\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}$ 的特征具有负实部。将式（4-2）代入式（4-3），得

$$J = \int_0^\infty (\boldsymbol{x}^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{x}^{\mathrm{T}}\boldsymbol{K}^{\mathrm{T}}\boldsymbol{R}\boldsymbol{K}\boldsymbol{x})\,\mathrm{d}t$$

$$= \int_0^\infty \boldsymbol{x}^{\mathrm{T}}(\boldsymbol{Q} + \boldsymbol{K}^{\mathrm{T}}\boldsymbol{R}\boldsymbol{K})\boldsymbol{x}\,\mathrm{d}t$$

设对任意 $\boldsymbol{x}$ 都有

$$\boldsymbol{x}^{\mathrm{T}}(\boldsymbol{Q} + \boldsymbol{K}^{\mathrm{T}}\boldsymbol{R}\boldsymbol{K})\boldsymbol{x} = -\frac{\mathrm{d}}{\mathrm{d}t}(\boldsymbol{x}^{\mathrm{T}}\boldsymbol{P}\boldsymbol{x})$$

式中，$\boldsymbol{P}$ 为正定实对称矩阵，则可导出

$$\boldsymbol{x}^{\mathrm{T}}\ (\boldsymbol{Q} + \boldsymbol{K}^{\mathrm{T}}\boldsymbol{R}\boldsymbol{K})\ \boldsymbol{x} = -\dot{\boldsymbol{x}}^{\mathrm{T}}\boldsymbol{P}\boldsymbol{x} - \boldsymbol{x}^{\mathrm{T}}\boldsymbol{P}\dot{\boldsymbol{x}}$$

$$= -\boldsymbol{x}^{\mathrm{T}}\ [\ (\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K})^{\mathrm{T}}\boldsymbol{P} + \boldsymbol{P}\ (\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K})\]\ \boldsymbol{x}$$

由 Lyapunov 第二方法可知，对于给定的正定矩阵 $\boldsymbol{Q} + \boldsymbol{K}^{\mathrm{T}}\boldsymbol{R}\boldsymbol{K}$，如果矩阵 $\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}$ 是稳定的，则存在正定矩阵 $\boldsymbol{P}$，使得

$$(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K})^{\mathrm{T}}\boldsymbol{P} + \boldsymbol{P}(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}) = -(\boldsymbol{Q} + \boldsymbol{K}^{\mathrm{T}}\boldsymbol{R}\boldsymbol{K}) \tag{4-4}$$

成立，由此可求出 J 为

$$J = \int_0^\infty \boldsymbol{x}^{\mathrm{T}}(\boldsymbol{Q} + \boldsymbol{K}^{\mathrm{T}}\boldsymbol{R}\boldsymbol{K})\boldsymbol{x}\,\mathrm{d}t = -\boldsymbol{x}^{\mathrm{T}}\boldsymbol{P}\boldsymbol{x}\Big|_0^\infty$$

$$= -\boldsymbol{x}^{\mathrm{T}}(\infty)\boldsymbol{P}\boldsymbol{x}(\infty) + \boldsymbol{x}^{\mathrm{T}}(0)\boldsymbol{P}\boldsymbol{x}(0)$$

由于假定 $\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}$ 的所有特征值都具有负实部，因此有 $\boldsymbol{x}(\infty) \to 0$。这样可得到

$$J = \boldsymbol{x}^{\mathrm{T}}(0)\boldsymbol{P}\boldsymbol{x}(0) \tag{4-5}$$

即 J 可由初始条件 $\boldsymbol{x}$（0）和 $\boldsymbol{P}$ 得到。

由于 $\boldsymbol{R}$ 为正定实对称矩阵，因此可将 $\boldsymbol{R}$ 写成

$$\boldsymbol{R} = \boldsymbol{T}^{\mathrm{T}}\boldsymbol{T}$$

式中，$\boldsymbol{T}$ 为一非奇异矩阵。这样式（4-4）可写成

$$(\boldsymbol{A}^{\mathrm{T}} - \boldsymbol{K}^{\mathrm{T}}\boldsymbol{B}^{\mathrm{T}})\boldsymbol{P} + \boldsymbol{P}(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}) + \boldsymbol{Q} + \boldsymbol{K}^{\mathrm{T}}\boldsymbol{T}^{\mathrm{T}}\boldsymbol{T}\boldsymbol{K} = 0$$

上式经整理还可写成

$$\boldsymbol{A}^{\mathrm{T}} + \boldsymbol{P}\boldsymbol{A} + [\boldsymbol{T}\boldsymbol{K} - (\boldsymbol{T}^{\mathrm{T}})^{-1}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{P}]^{\mathrm{T}}[\boldsymbol{T}\boldsymbol{K} - (\boldsymbol{T}^{\mathrm{T}})^{-1}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{P}] - \boldsymbol{P}\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{P} + \boldsymbol{Q} = 0$$

即有　　$\boldsymbol{x}^{\mathrm{T}}[\boldsymbol{T}\boldsymbol{K} - (\boldsymbol{T}^{\mathrm{T}})^{-1}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{P}]^{\mathrm{T}}[\boldsymbol{T}\boldsymbol{K} - (\boldsymbol{T}^{\mathrm{T}})^{-1}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{P}]\boldsymbol{x}$

由于上式是非负的，其最小值为零，即

$$\boldsymbol{TK} = (\boldsymbol{T}^{\mathrm{T}})^{-1}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{P}$$

因此可求得 J 取极小值时的最优反馈增益矩阵为

$$\boldsymbol{K} = \boldsymbol{T}^{-1}(\boldsymbol{T}^{\mathrm{T}})^{-1}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{P} = \boldsymbol{R}^{-1}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{P} \tag{4-6}$$

最优控制 $\boldsymbol{u}(t)$ 为

$$\boldsymbol{u}(t) = -\boldsymbol{K}\boldsymbol{x}(t) = -\boldsymbol{R}^{-1}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{P}\boldsymbol{x}(t)$$

式（4-6）中的矩阵 $\boldsymbol{P}$ 必须满足式（4-4）或者满足方程

$$\boldsymbol{A}^{\mathrm{T}}\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} - \boldsymbol{P}\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{P} + \boldsymbol{Q} = 0 \tag{4-7}$$

式（4-7）称为代数 Riccati 方程。

综上所述，系统的设计步骤可概括如下：

1）求解式（4-7）Riccati 方程，求得矩阵 $\boldsymbol{P}$。如果正定矩阵 $\boldsymbol{P}$ 存在（对某些系统，可能不存在），则系统是稳定的或矩阵 $\boldsymbol{A}-\boldsymbol{B}\boldsymbol{K}$ 是稳定的。

2）将此矩阵 $\boldsymbol{P}$ 代入方程式（4-6），得到的即为最优反馈增益矩阵 $\boldsymbol{K}$。

（二）MATLAB（控制系统工具箱）实现

MATLAB 控制系统工具箱中提供了完整的解决线性二次型最优控制问题的命令及算法，其中命令 lqr 和 lqry 可以直接求解二次型调节器问题以及相关的 Riccati 方程。这两个命令的格式如下：

[K, P, E] = lqr (A, B, Q, R, N)

[K, P, E] = lqry (A, B, C, D, Q, R)

式中，K 为最优反馈增益矩阵；P 为对应 Riccati 方程的唯一正定解（若矩阵 $\boldsymbol{A}-\boldsymbol{B}\boldsymbol{K}$ 是稳定矩阵，则总有 $\boldsymbol{P}$ 的正定解存在）；E 为 $\boldsymbol{A}-\boldsymbol{B}\boldsymbol{K}$ 的特征值；N 为可选项，其代表交叉乘积项的加权矩阵。lqry 命令用于求解二次调节器问题的特例，即目标函数中用输出 $\boldsymbol{y}$ 来代替状态 $\boldsymbol{x}$，则目标函数为

$$J = \int_0^{\infty}(\boldsymbol{y}^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{y} + \boldsymbol{u}^{\mathrm{T}}\boldsymbol{R}\boldsymbol{u})\,\mathrm{d}t$$

下面的讨论可进一步说明之。

（三）问题讨论——一阶直线倒立摆系统的线性二次型最优控制

1. 理论设计

（1）系统的状态空间描述　由第二章第三节知：一阶直线倒立摆系统状态空间描述方程为

$$\boldsymbol{X}^{\cdot} = \begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\\ \dot{x}_4\end{bmatrix} = \begin{bmatrix}0 & 1 & 0 & 0\\ 40 & 0 & 0 & 0\\ 0 & 0 & 0 & 1\\ -6 & 0 & 0 & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\\ x_4\end{bmatrix} + \begin{bmatrix}0\\ -2\\ 0\\ 0.8\end{bmatrix}u = \boldsymbol{A}\boldsymbol{X} + \boldsymbol{B}u$$

$$\boldsymbol{Y} = \begin{bmatrix}\theta\\ x\end{bmatrix} = \begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 0 & 1 & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\\ x_4\end{bmatrix} = \boldsymbol{C}\boldsymbol{X}$$

其中：$\boldsymbol{X}=[\theta,\ \dot{\theta},\ x,\ \dot{x}]^{\mathrm{T}}$；$u=F$　$\boldsymbol{Y}=[\theta,\ x]^{\mathrm{T}}$。

（2）系统可控制性和可观测性的判定　“可控性与可观性是现代控制理论的重要贡献之一”。我们事先确定“所研究问题的可控制性”，有助于使我们避免对一些“不可控制的问题”进行徒劳的工作；而对“所研究问题可观测性”的判断，有助于我们了解“能否利用系统有限的可测输出量来求取系统的全部状态变量”，进而实现对系统实施“全状态反馈控制”。

1）可控制性的判定。从上述系统状态空间描述可知，系统阶数为 $n=4$，控制量为 U，即 $M=1$，所以，可控性矩阵 $\boldsymbol{V}$ 为 4×4 矩阵，$\boldsymbol{V}$ 的秩数是 4（系统可控制的充要条件）。

矩阵 $\boldsymbol{V}$ 可由下式求得

$$\boldsymbol{V}=[\boldsymbol{B}\quad \boldsymbol{AB}\quad \boldsymbol{A}^2\boldsymbol{B}\quad \boldsymbol{A}^3\boldsymbol{B}]$$

其中：

$$\boldsymbol{B}=\begin{bmatrix}0\\-2\\0\\0.8\end{bmatrix}\quad \boldsymbol{AB}=\begin{bmatrix}-2\\0\\0.8\\0\end{bmatrix}\quad \boldsymbol{A}^2\boldsymbol{B}=\begin{bmatrix}0\\-80\\0\\12\end{bmatrix}\quad \boldsymbol{A}^3\boldsymbol{B}=\begin{bmatrix}-80\\0\\12\\0\end{bmatrix}$$

因此，$\boldsymbol{V}$ 可表示为

$$\boldsymbol{V}=\begin{bmatrix}0&-2&0&-80\\-2&0&-80&0\\0&0.8&0&12\\0.8&0&12&0\end{bmatrix}$$

这个 4×4 矩阵的秩数是 4，因为该矩阵所有的列都是相互独立的，因此就可确认该系统是可控制的。

对于“一阶倒立摆系统的可控性问题”，尽管我们依据经验或直觉也可以判断出来（例如：人用手指尖顶一木杆现象），但是，对于有些问题仅凭直觉就不然了，例如“一阶双摆控制问题”（参见本章习题 4-14）。

2）系统可观测性的判定。因为 $n=4$，系统的输出有两个（角度 θ 和位移 x）所以 $L=2$，可观测性矩阵 $\boldsymbol{S}$ 的阶数为 8×4，表示为

$$\boldsymbol{S}=\begin{bmatrix}\boldsymbol{C}\\\boldsymbol{CA}\\\boldsymbol{CA}^2\\\boldsymbol{CA}^3\end{bmatrix}\quad \text{其中 } \boldsymbol{C}=\begin{bmatrix}1&0&0&0\\0&0&1&0\end{bmatrix}$$

$\boldsymbol{CA}$、$\boldsymbol{CA}^2$、$\boldsymbol{CA}^3$ 由下式给出：

$$\boldsymbol{CA}=\begin{bmatrix}0&1&0&0\\0&0&0&1\end{bmatrix}\quad \boldsymbol{CA}^2=\begin{bmatrix}40&0&0&0\\-6&0&0&0\end{bmatrix}\quad \boldsymbol{CA}^3=\begin{bmatrix}0&40&0&0\\0&-6&0&0\end{bmatrix}$$

因此，可观测性矩阵 S 如下：

$$S = \begin{bmatrix} 1.0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1.0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 40 & 0 & 0 & 0 \\ -6 & 0 & 0 & 0 \\ 0 & 40 & 0 & 0 \\ 0 & -6 & 0 & 0 \end{bmatrix}$$

显见可观测性矩阵 S 的前 4 行相互独立，因此可观测性矩阵 S 的秩数是 4，与系统阶数 $n=4$ 一致，所以系统是可观测的。

下面，我们再利用 MATLAB 语言求解一阶倒立摆系统的可控性与可观性：

采用状态变量反馈的控制系统，其能控性和能观测性可以用 MATLAB 的 ctrb 函数和 obsv 函数来直接加以检验。能控性函数 ctrb 的输入是系统矩阵 A 和输入矩阵 B，而输出则是能控性矩阵 P；同理，obsv 函数的输入是系统矩阵 A 和输出矩阵 C，而输出则是能观性矩阵 Q。输入如下语句：

```
P = ctrb(A,B); % 计算能控性矩阵
N = rank(P); % 计算能控性矩阵的秩
```

如果 N = n = 4，则证明系统为能控；否则，系统为不能控。

```
Q = obsv(A,C); % 计算能观测性矩阵
N = rank(Q); % 计算能观测性矩阵的秩
```

如果 N = n = 4，则证明系统为能观测；否则，系统为不能观测。

通过已知的 A、B 和 C，可以证明：系统是能控和能观测的。

综上所述，该一阶倒立摆系统是可控制的，同时也是可观测的。

(3) 线性二次型最优控制系统设计　对于一阶倒立摆系统，由于控制量为单一的 U，即 R 为一阶矩阵，我们可取 $R=1$；对于 Q，我们取

$$Q = \mathrm{diag}[q_{11} \quad q_{22} \quad q_{33} \quad q_{44}]$$

可见，对于任意一组 Q 中的参数，我们都可以得到其对应的最优反馈控制量。

当 q_{11}，q_{22}，q_{33}，q_{44} 较大时，控制量权重相对较小，将需要较大作用力。根据不同的要求，可取不同 Q 矩阵，这里我们取 $q_{11}=q_{22}=200$，$q_{33}=q_{44}=50$，则由 MATLAB 中提供的解决线性二次最优控制问题的命令 K = lqr (A, B, Q, R)，可得：K = (121.31, 12.12, 5.03, 7.67)，从而有如下控制方程和图 4-40 所示的控制系统结构图。

$$F = -[121.31 \quad 12.12 \quad 5.03 \quad 7.67]X + r$$

图 4-40 所示的全状态反馈控制系统在实际实现中存在着“θ 与 $\dot{x}$”测量不准问题（这是因为“微分”作用对信号噪声非常敏感，而使得 $\dot{\theta}$ 与 $\dot{x}$ 的检测结果不可用）。因此，还有如图 4-41 所示的基于状态观测器的一阶倒立摆控制方案。

有关状态观测器的设计与应用问题，受篇幅所限此处就不再赘述了。

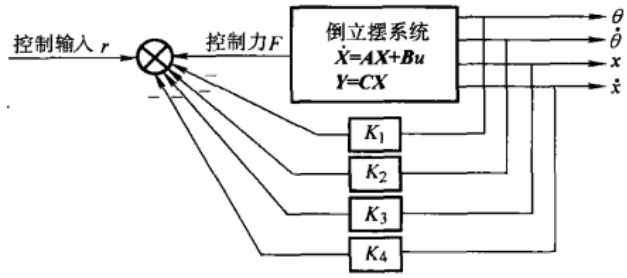


图 4-40　控制系统结构图

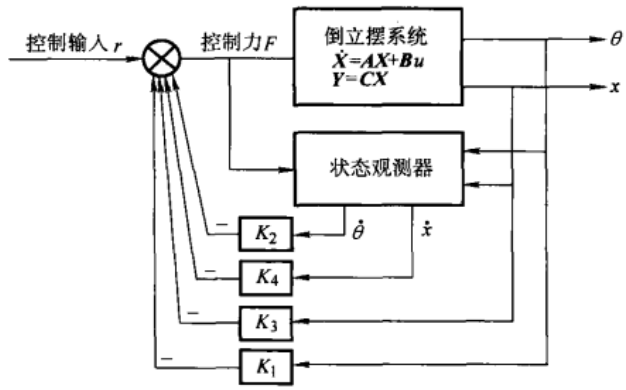


图 4-41　基于状态观测器的一阶倒立摆控制方案

2. 仿真实验

（1）基于系统状态空间描述的仿真

1）系统仿真结构如图 4-42 所示。

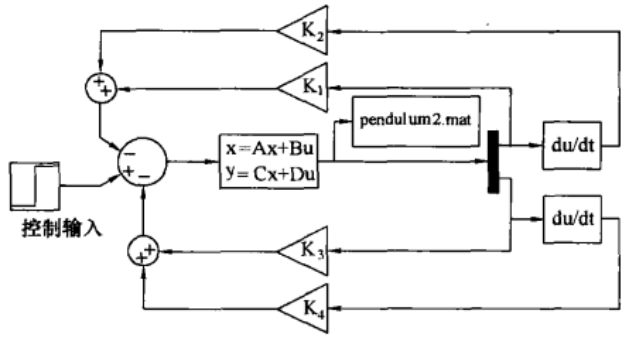


图 4-42　系统仿真结构图

2）仿真程序：

```
% Inverted pendulum2
% singnals recuperation
% 将导入到 pendul2. mat 中的仿真实验数据读出
load pendul2. mat
t = signals (1,:);
q = signals (2,:);
```

```
x = signals (3,:);
% drawing x(t) and theta(t) response signals
% 画摆杆线速度和摆角角速度的曲线图
% 定义图形名称、标题、横纵坐标名和坐标范围，画出曲线
figure(1)
hf = line(t,q(:));
xlabel('Time(s)')
ylabel('Angle evolution(rad)')
axis([0 10 -0.1 0.3])
axet = axes('Position',get(gca,'Position'),...
            'XAxisLocation','bottom',...
            'YAxisLocation','right','Color','None',...
            'XColor','k','YColor','k');
ht = line(t,x,'color','r','parent',axet);
ylabel('Evolution of the x position(m)')
axis([0 10 -0.1 0.3])
title('\theta(t) and x(t) Response to a step input')
gtext('\leftarrow x(t)'),gtext('\theta(t) \uparrow')
```

3）仿真结果如图 4-43 所示。

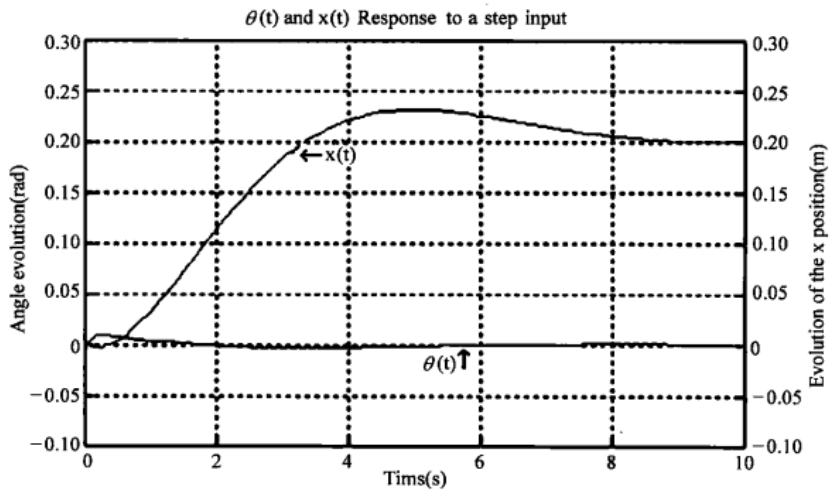


图 4-43 系统仿真结果

（2）基于系统精确模型的仿真 以系统精确模型代替图 4-42 中的近似模型（状态空间描述），重新进行仿真实验。

1）系统仿真结构如图 4-44 所示。

2）仿真结果如图 4-45 所示。

（3）鲁棒性实验 为了进一步检验控制器的鲁棒性能，不妨改变倒立摆的摆杆质量和长度多做几组实验，同时为了便于比较，将几个仿真结果绘在同一坐标下，实验的仿真结果如图 4-46 和图 4-47 所示。

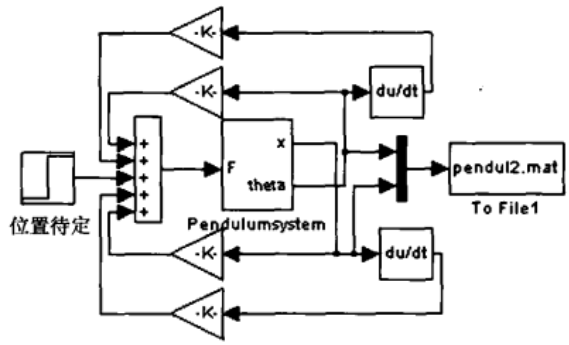


图 4-44　系统仿真结构图

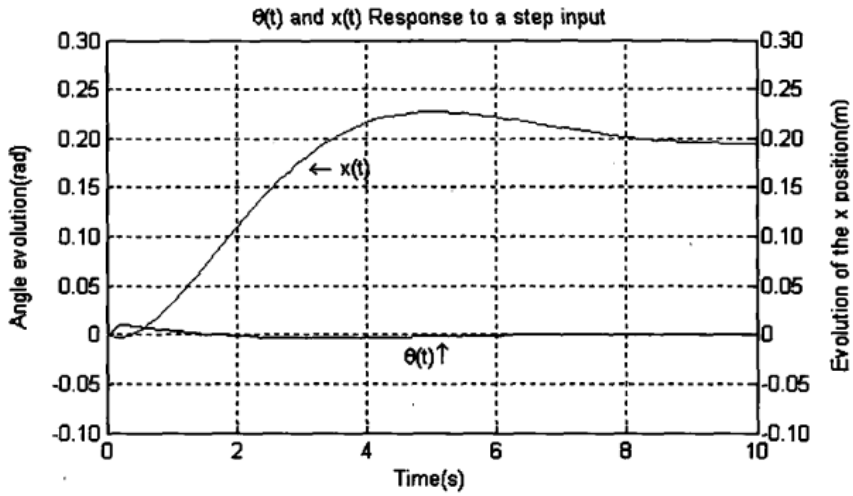


图 4-45　以精确模型为控制对象的仿真结果图

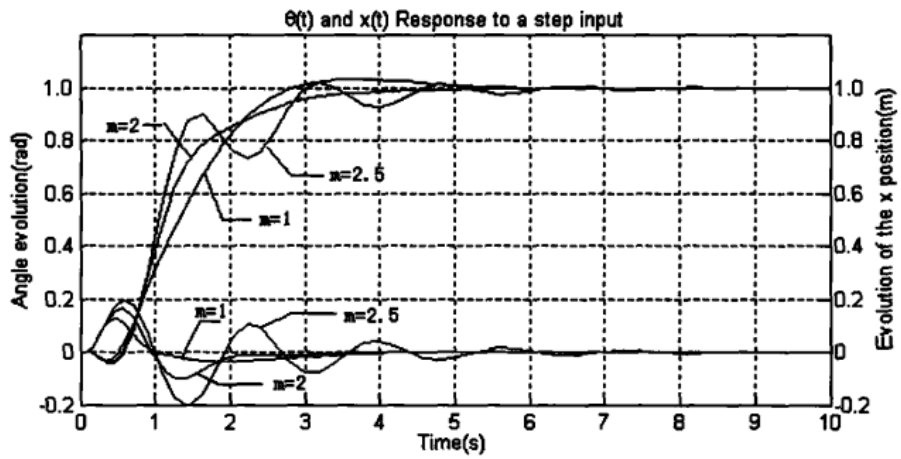


图 4-46　摆长不变摆杆质量变化的一组曲线

3. 结论

1）线性二次型最优控制策略通过全状态反馈控制的方式，可以同时达到小车位置伺服控制和摆角控制的目的，并实现了系统动态性能的最优。

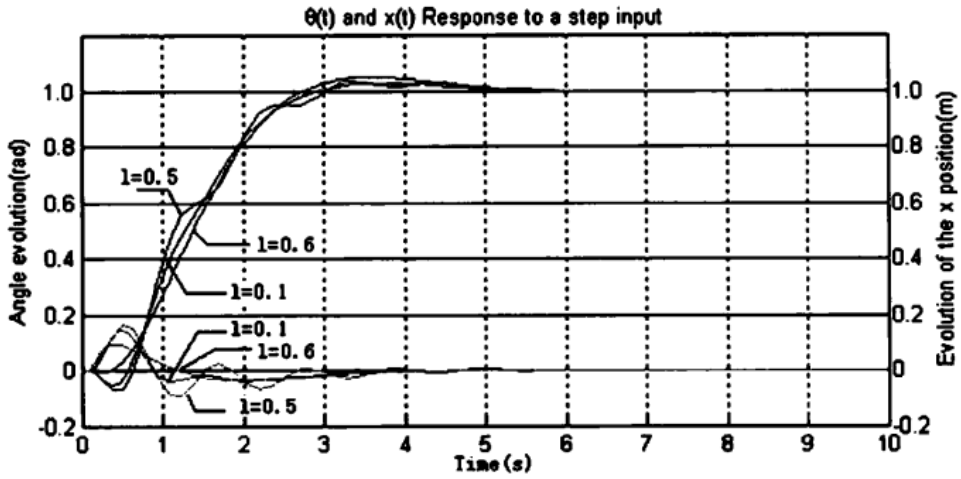


图 4-47 摆杆质量不变摆长变化的一组曲线

2）从图 4-43 与图 4-45 的仿真结果可见，近似模型与精确模型的动态响应是很相近的，这也从另一方面说明，前面所述的模型简化是合理的。

3）系统具有较好的鲁棒性，具体表现在系统对摆长与摆杆质量两参数的大范围变化（3～4 倍）表现出较强的不敏感性。

二、模型参考自适应控制系统设计

在经典控制理论中要设计一个性能良好的反馈控制系统，通常需要掌握被控对象的数学模型，然而，实际上有一些被控对象的数学模型事先难以获得，或者模型的参数是经常变化的。对于这样的被控对象，一般的控制方法往往难以得到满意的性能。如果系统本身能够不断地测量被控对象的性能或参数，并把系统当前的运行指标与期望的性能指标相比较，进而改变控制器的结构或参数，就可以使系统运行在某种意义下的最优或次优状态。按照这样的思想建立的控制系统，称为自适应控制系统。

自适应控制的思想于 20 世纪 50 年代提出，当时主要针对具体的设计方案进行讨论，没有形成一个理论体系。20 世纪 60 年代现代控制理论发展所取得的一些成果，例如状态空间法、稳定性理论、最优控制、随机控制、参数估计等，为自适应控制理论的形成和发展准备了条件。20 世纪 70 年代以来自适应控制理论的研究有了显著的进展，提出了一系列的控制方案，并出现了许多实际应用的例子。微处理器的迅猛发展一方面为自适应控制的应用创造了技术条件，另一方面也促进了自适应控制理论的发展。

（一）自适应控制系统的构成

根据设计原理和结构的不同，自适应控制系统可分为如下几种形式：

1. 变增益自适应控制系统

这种系统的结构如图 4-48 所示。当控制器的参数随工作状况和环境的变化而变化时，通过测量系统的某些变量，按照规定的程序可改变控制器的增益结构。这种系统具有结构简单和响应迅速的特点，但是，难以完全克服系统参数变化带来的影响。

2. 模型参考自适应控制系统

模型参考自适应控制系统的基本结构如图 4-49 所示。它由两个环路组成，内环由调节

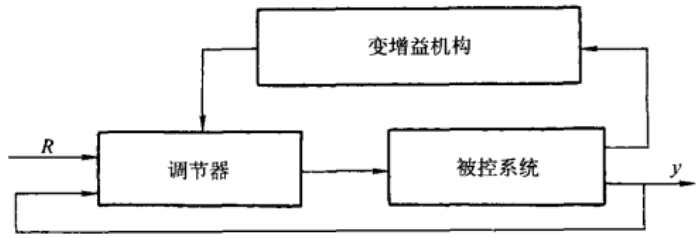


图 4-48　变增益自适应控制

器和被控对象组成可调系统，外环由参考模型与自适应机构组成。参考模型代表某个优化指标。当被控对象受干扰影响使运行特性偏离了最优轨迹时，参考模型的输出与被控对象的输出相比较产生了一个广义误差 e。该误差通过自适应机构，根据一定的自适应规律产生反馈作用，以修正调节器的参数或产生一个辅助的控制信号，促使可调系统跟踪参考模型，使广义误差趋于极小值。

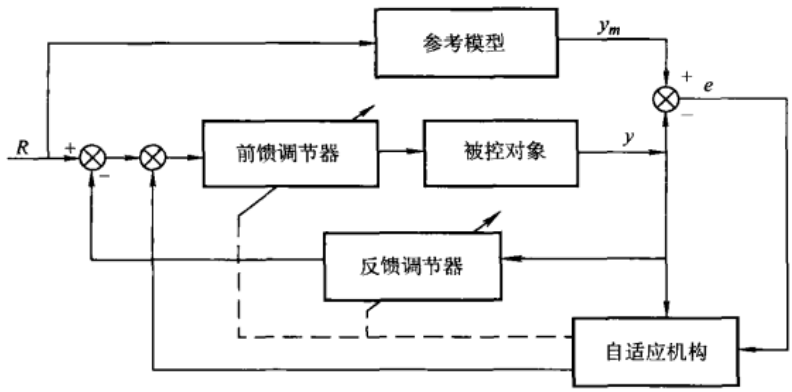


图 4-49　模型参考自适应控制系统

模型参考自适应控制设计方法主要依据李亚普诺夫稳定性理论和超稳定理论，可以适用于线性控制系统和非线性控制系统，这样可以保证系统在稳定的前提下对系统的参数变化具有适应性，并提高有关的性能指标。

3. 自校正控制系统

自校正控制系统也称为参数自适应系统。其一般结构如图 4-50 所示。它也有两个环路，一个环路由调节器和被控对象组成，类似于通常的反馈控制系统；另一个环路由递推参数辨识器与调节器参数设计计算机组成。自校正控制系统将在线参数辨识与调节器的设计有机地结合在一起。在运行过程中，首先进行被控对象的参数辨识，然后根据参数辨识的结果进行调节器参数的选择设计，并根据设计结果修正调节器参数以达到有效地消除被控对象参数扰动所造成的影响。

（二）卫星跟踪望远镜模型参考自适应控制系统

下面要研究的对象是美国国家航空航天局的 24in（1in = 0.0254m）光学跟踪望远镜，它用于激光测距和光学通信实验。由于激光束很窄，因此系统对动态跟踪精度要求很高，位置误差应小于 0.16in · s。望远镜工作于三种方式：搜索方式、自动跟踪方式和程序跟踪方式。一般来说，在运行过程中，由于受到望远镜仰角的变化以及过盈量轴承的采用（以保证系

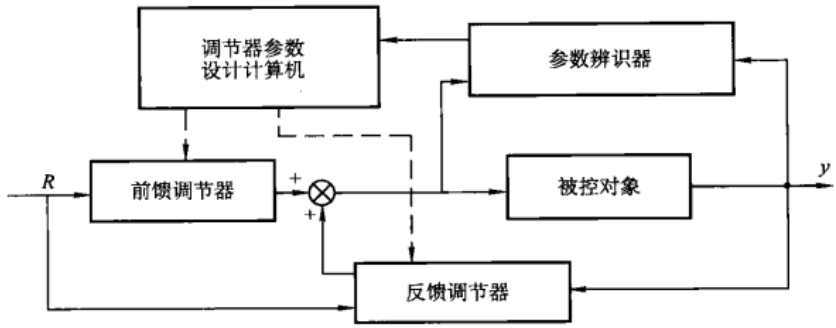


图 4-50 自校正控制系统

统具有较高的回转精度），负载惯量、转矩和动静摩擦发生变化是不可避免的，因此采用常规的控制很难完成上述三个任务，于是考虑采用模型参考自适应控制方法。

望远镜的伺服控制系统如图 4-51 所示，其中 u 为自适应控制信号，f 代表主轴系统摩擦力的特性，它与机构的运行速度有明显的非线性关系，其关系如图 4-52 所示，也称之为库仑摩擦。

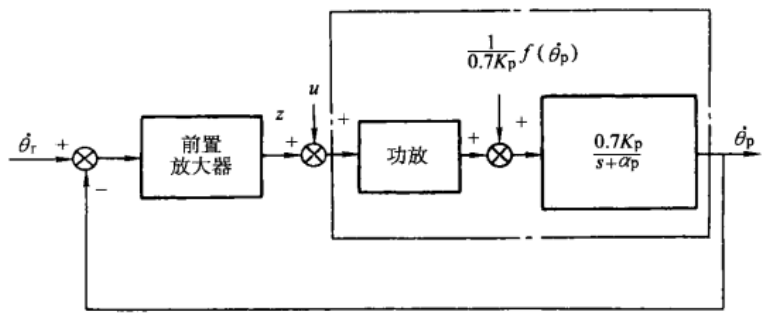


图 4-51 望远镜伺服控制系统结构图

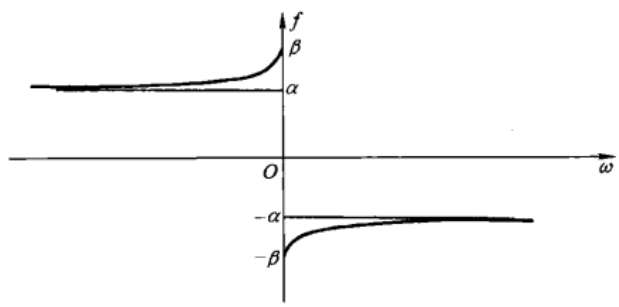


图 4-52 摩擦力与速度的非线性关系

根据模型参考自适应理论[34-35]，可以得到自适应控制信号 u 的形式为

$$u = K_1\theta_p + K_2 z + K_3 \mathrm{sgn}\dot{\theta}_p$$

$$K_1(t) = B_1\int_0^t e\dot{\theta}_p \mathrm{d}t + C_1 e\dot{\theta}_p$$

$$K_2(t) = B_2 \int_0^t \dot{e}z\mathrm{d}t + C_2\dot{e}z$$

$$K_3(t) = B_3 \int_0^t \dot{e}\mathrm{sgn}\dot{\theta}_p\mathrm{d}t + C_3\dot{e}\mathrm{sgn}\dot{\theta}_p$$

式中，K_1、K_2、K_3 为可调参数，它们由自适应机构调节；第三项($K_3(t)$)用来补偿轴承摩擦(库仑摩擦)的影响。

(三) 仿真实验研究

下面以程序跟踪方式为例介绍模型参考自适应控制在卫星跟踪望远镜位置伺服系统中的具体应用。系统首先根据已知的数据对卫星未来轨线作出预测，再把预测信号经前馈补偿器加到指令信号中去，图 4-53 给出了程序跟踪自适应控制方案的仿真结构。

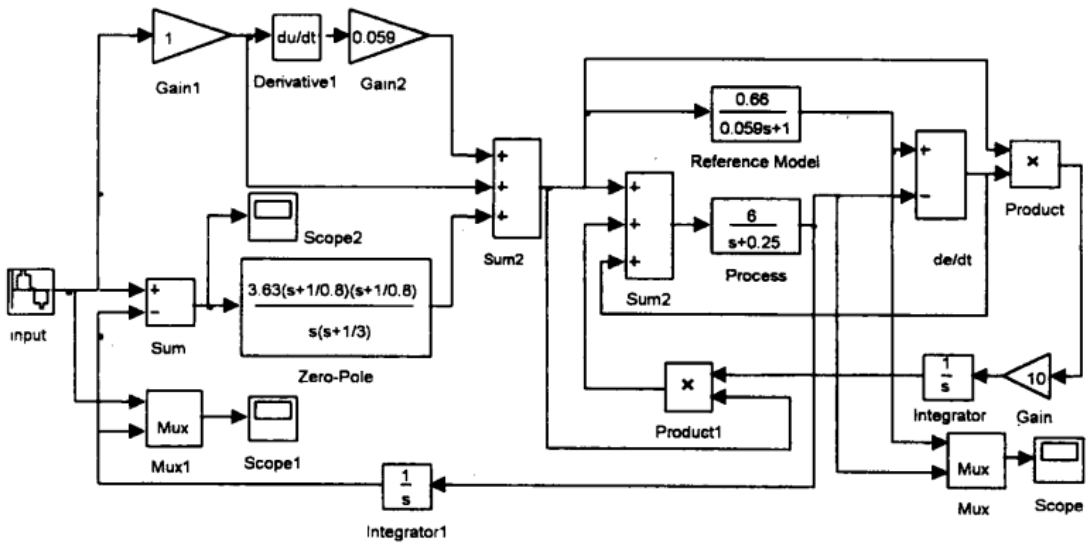


图 4-53　程序跟踪自适应控制方案

1. 常规控制策略时望远镜的位置跟踪特性

对于图 4-53 所示系统，将自适应环节去掉，控制器采用 PID 结构，则仿真结果如图 4-54 所示，从中可见，系统跟踪有误差，而且在过零点处有抖动。这一结果显然是我们所不希望的，应该解决之。

2. 采用自适应控制策略处理这一问题

方案如图 4-53 所示。

1) 设参考位置输入为正弦信号，角频率为 0.2rad/s，幅值为 1.0rad/s，对象传递函数为 $6/(s+0.25)$。通过仿真可得如下位置跟踪曲线(见图 4-55)、位置跟踪误差曲线(见图 4-56)及速度跟踪曲线(见图 4-57)。从上述仿真结果可见，系统动态跟踪特性明显得到改善。

2) 比较对象参数变化对系统性能的影响。设参考位置输入不变，对象传递函数变为 $4/(s+0.4)$。仿真得如下位置跟踪曲线(见图 4-58)、位置跟踪误差曲线(见图 4-59)及速度跟踪曲线(见图 4-60)。从中可见，对象模型参数的变化对系统动态跟踪性能基本上没有影响。

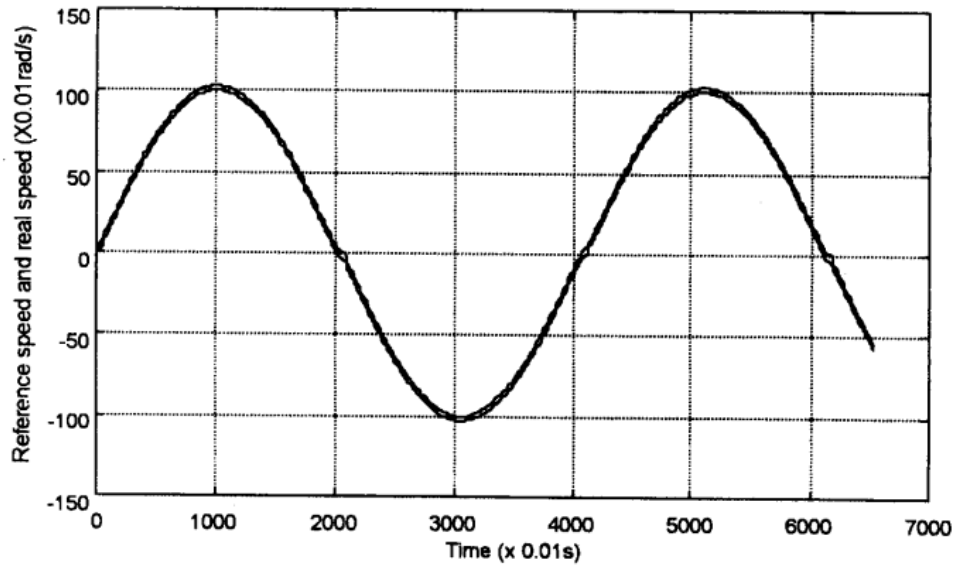


图 4-54 常规控制策略时的仿真结果

3. 仿真实验

从上述的仿真实验结果可以看出，模型参考自适应控制方法对系统模型的依赖性要低于常规的控制方法，对于慢变化的参数时变系统和某些非线性问题具有良好的鲁棒性；同时，它能够较好地克服低速情况下非线性摩擦(库仑摩擦)对控制系统带来的影响。当然，自适应控制系统的复杂程度要高于常规的控制方法，具体应用时还要考虑许多工程问题。

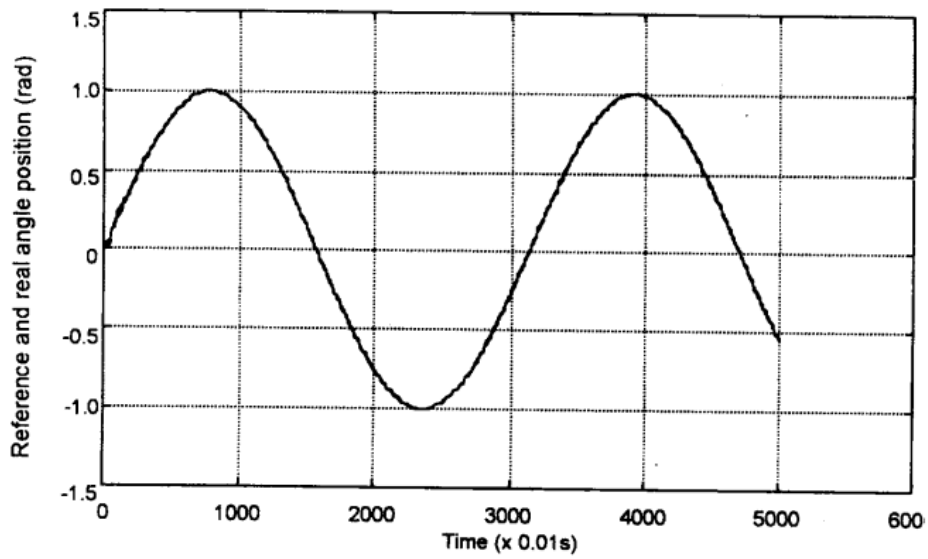


图 4-55 正弦输入信号的位置跟踪曲线 (一)

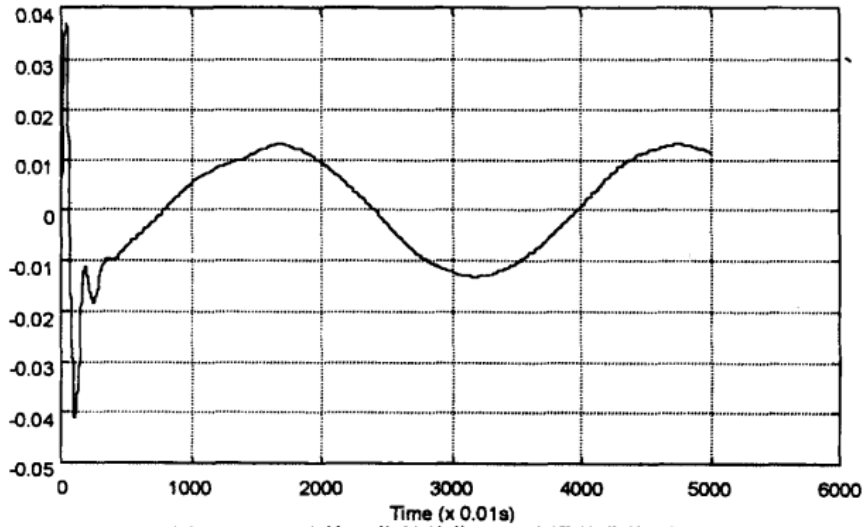


图 4-56　正弦输入信号的位置跟踪误差曲线（一）

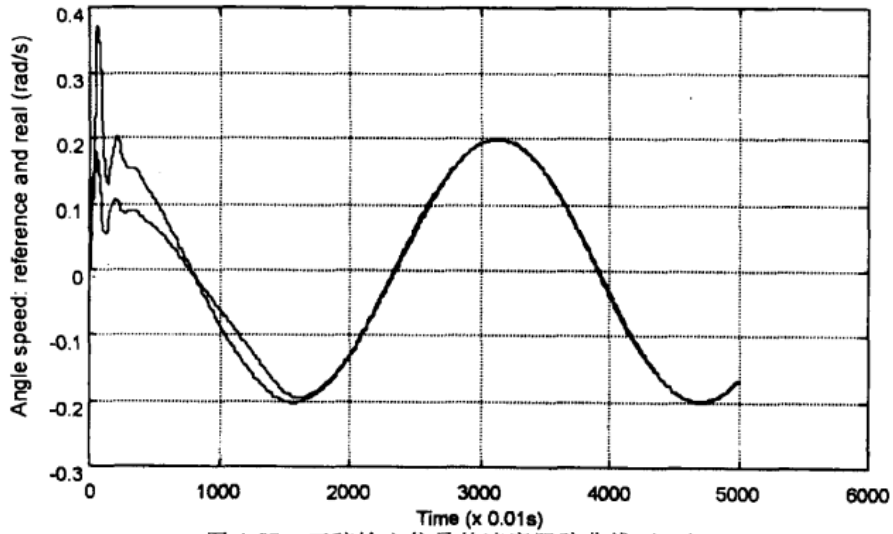


图 4-57　正弦输入信号的速度跟踪曲线（一）

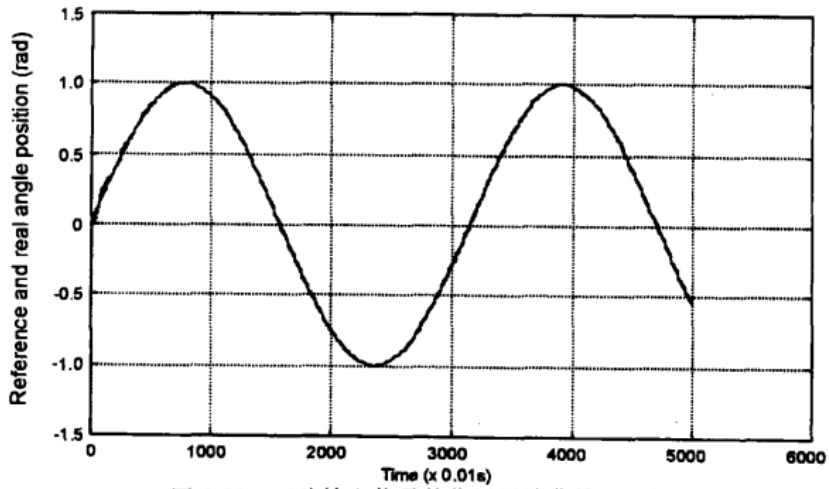


图 4-58　正弦输入信号的位置跟踪曲线（二）

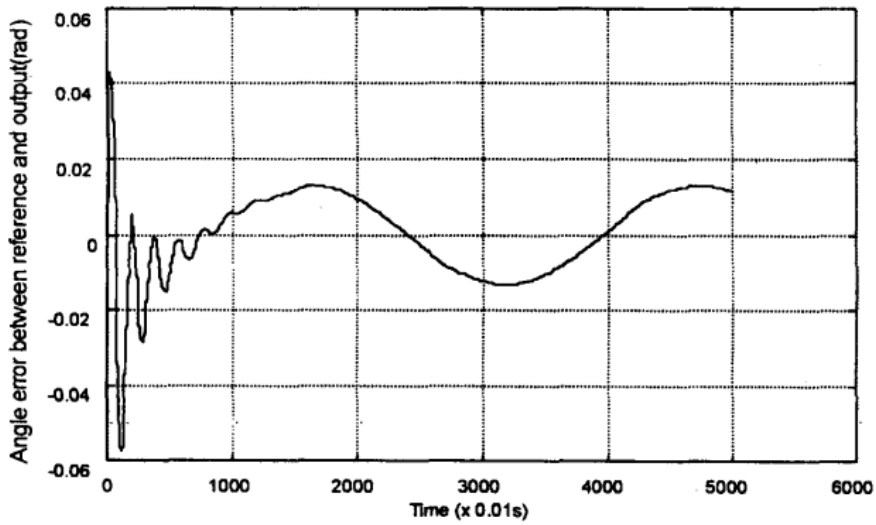


图 4-59 正弦输入信号的位置跟踪误差曲线（二）

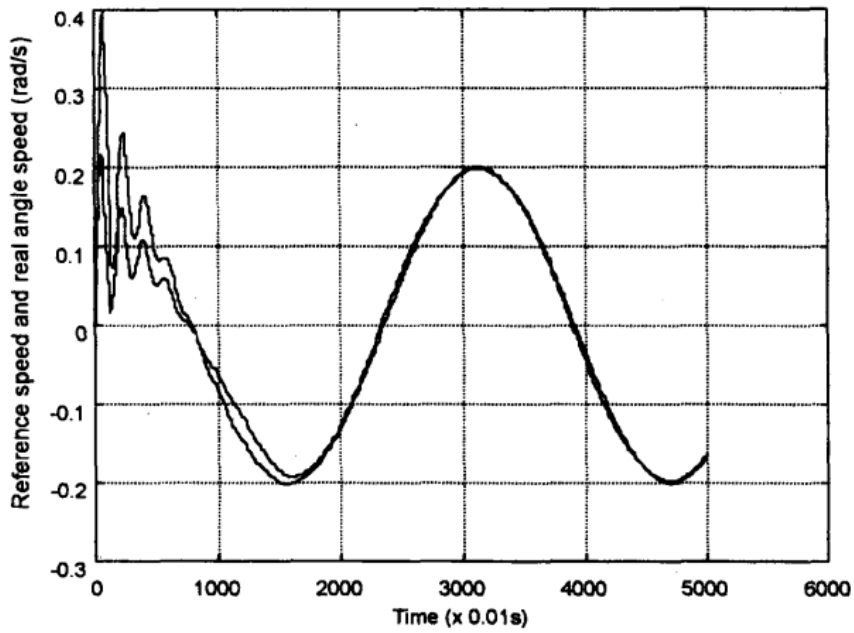


图 4-60 正弦输入信号的速度跟踪曲线（二）

第五节 基于时间最优控制的起重机防摆控制技术研究

一、问题的提出

在图 4-61 所示的起重机系统中，为使重物尽快地由 A 点移动至 B 点（同时，还要保证重物在抵达 B 点处时满足 $|\theta| \leqslant \Delta$ 条件），我们希望通过控制量 $u(t)$ 适当控制小车的运行加速度 $\ddot{x}$ 来实现这一目标。

显然，这是一个最优控制问题。本节我们将借助于 Simulink 封装技术（子系统），在模型验证的基础上，采用时间最优控制方案，实现起重机防摆控制的系统仿真实验。

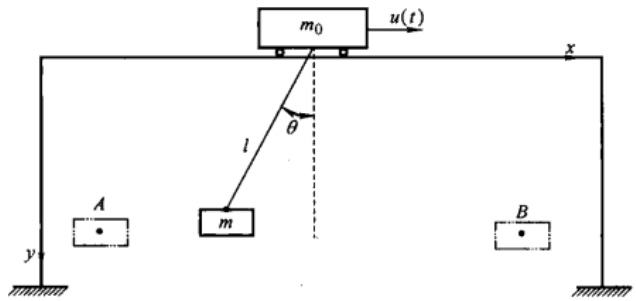


图 4-61　起重机系统原理图

二、时间最优控制[55]

最优控制问题，是我们在从事的研究工作中经常碰到的问题。采取什么手段能够实现以最小的代价换取最大的收益是最优控制要研究的主要问题。

维纳（Wiener）在 1940 年提出了相对于某个性能指标进行设计的概念；1950 年，米顿纳尔（Medonel）首先将这个概念用于研究继电器系统在单位阶跃激励作用下过渡过程时间最短的控制问题；后来，庞特里亚金（Понтрягин，Л. С.）、贝尔曼（Bellman）等人根据大量的工程实际问题（如航空、航天、飞行器控制等问题），总结并提出了按某一给定的性能指标设计最优控制系统的理论——“极大值原理”、“动态规划”。

“如何将已有的控制理论应用于工程实际问题，是我们从事自动化技术研究与应用人员的任务”。

（一）时间最优控制问题的数学描述

设受控系统的状态方程为

$$\begin{cases}\dot{x}(t)=f[x(t),u(t),t], & x\in R^n\\ x(t_0)=x_0\end{cases}$$

给定的终端状态为 $x(t_f)=x_f$，控制变量的约束为 $u(t)\in U\in R^r$。

若欲选择允许的最优控制函数 $u^*(t)$，其将使得受控系统的状态方程从初始状态 x_0 出发，转移到终端状态 x_f 的过程所需要的时间最短（即 $(t_f-t_0)=\min$），称这一问题为时间最优控制问题。

时间最优控制问题的性能指标函数可表示为

$$J=\int_{t_0}^{t_f}1\mathrm{d}t=t_f-t_0$$

（二）时间最优控制的必要条件

根据极大值原理可知，线性系统的时间最优控制有解的必要条件是，控制域 U 必须是“有界闭集”，且最优控制函数必定取这个闭集 U 的边界值。由此可得出时间最优的控制为

$$U=\left\{\boldsymbol{u}=\begin{bmatrix}u_1\\u_2\\\vdots\\u_r\end{bmatrix}\quad |u_i|\leqslant a_i,\ i=1,2,\cdots,r\right\}$$

由此可见，允许控制的时间最优控制系统是 Bang-Bang 型控制系统，只有继电型非线性系统才能实现时间最优控制。

（三）相平面分析法

对于如下方程所示的二阶系统，使用“状态平面图”分析系统运动状态比较方便。

$$\begin{cases}\dot{x}_1(t)=f_1[t,x_1(t),x_2(t)]\\ \dot{x}_2(t)=f_2[t,x_1(t),x_2(t)]\end{cases}$$

所谓“状态平面”，一般为二维平面，其水平轴记为 x_1，垂直轴记为 x_2。假设[$x_1(t)$，$x_2(t)$]表示该运动系统状态的一个解，则当 t 为固定值时，其“解”对应于状态平面上的一个点；当 t 连续变化时，其“解”在状态平面上形成的运动轨迹称为状态平面轨迹；当该二阶系统的形式满足 $\dot{x}_1(t)=x_2(t)$ 时，习惯上把这一特殊情况下的状态平面称为相平面，相应的状态平面轨迹称为相平面轨迹，或简称为相轨迹。

三、系统建模

由第二章第二节内容可知，定摆长起重机系统在（$D=\eta=0$）条件下的精确模型为

$$\begin{cases}\ddot{x}=\dfrac{F-(g\cos\theta+l\dot{\theta}^2)m\sin\theta}{m_0+m\sin^2\theta}\\ \ddot{\theta}=\dfrac{(F\cos\theta-ml\dot{\theta}^2\cos\theta\sin\theta-(m_0+m)g\sin\theta}{l(m_0+m\sin^2\theta)}\end{cases}$$

其在($\sin\theta\approx\theta,\cos\theta\approx1,\dot{\theta}^2\sin\theta\approx0$)近似条件下的简化模型如图 4-62 所示。

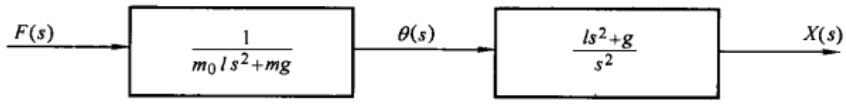


图 4-62 起重机系统动态结构图

系统在 $m_0=50\text{kg}$、$m=5\text{kg}$、$l=1\text{m}$、$g=9.8\text{m/s}^2$ 条件下的简化模型为

$$\begin{cases}F=50\ddot{x}+49\theta\\ \ddot{x}=\ddot{\theta}+9.8\theta\end{cases}$$

其等效动态结构图又可表示为图 4-63 所示的形式。

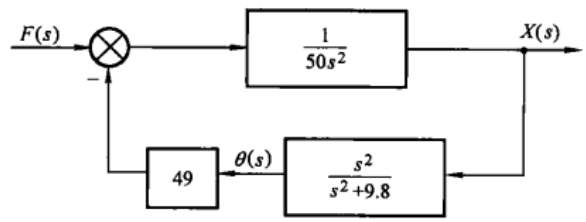


图 4-63 起重机系统动态结构图

四、模型验证

（一）模型封装

利用前面介绍的 Simulink 封装子系统功能，可使模型验证原理表示得更加简捷。如图 4-64 所示，上半部分为简化模型仿真图，下半部分为精确模型仿真图。

其中：

Fcn:(u[7]-9.8*u[8]*u[3]*u[4]-u[8]*u[6]*u[5]*u[5]*u[3])/(u[9]+u[8]*u[3]*u[3])

Fcn1:((u[7]-9.8*u[8]*u[3]*u[4]-u[8]*u[6]*u[5]*u[5]*u[3])/(u[9]+u[8]*u[3]*u[3]))*u[4]/u[6]-9.8*u[3]/u[6]

（二）模型验证

同本章第三节中的验证方法一样，下面用“必要条件法”来验证我们所建立的数学模型具备正确模型应具备的必要性质。

（1）实验设计　假定使起重机在($\theta=0,x=0$)初始状态下，突加一有限恒定作用力，则

依据经验知，小车位移将不断增大，而重物将在小车的一侧作往复摆动。这一结果可根据图 4-65 所示的原理予以说明：由于小车受到一恒定力的作用，因此初始状态 O 点为重物相对小车摆动的一个极限点，该恒定力的作用也将使得重物相对小车的摆动存在另一个摆动的极限点 A；同时，我们也知道，单摆运动的极限点为不稳定点，因此，在这一恒力作用的过程中，重物将在小车一侧的两极限点间作往复摆动。

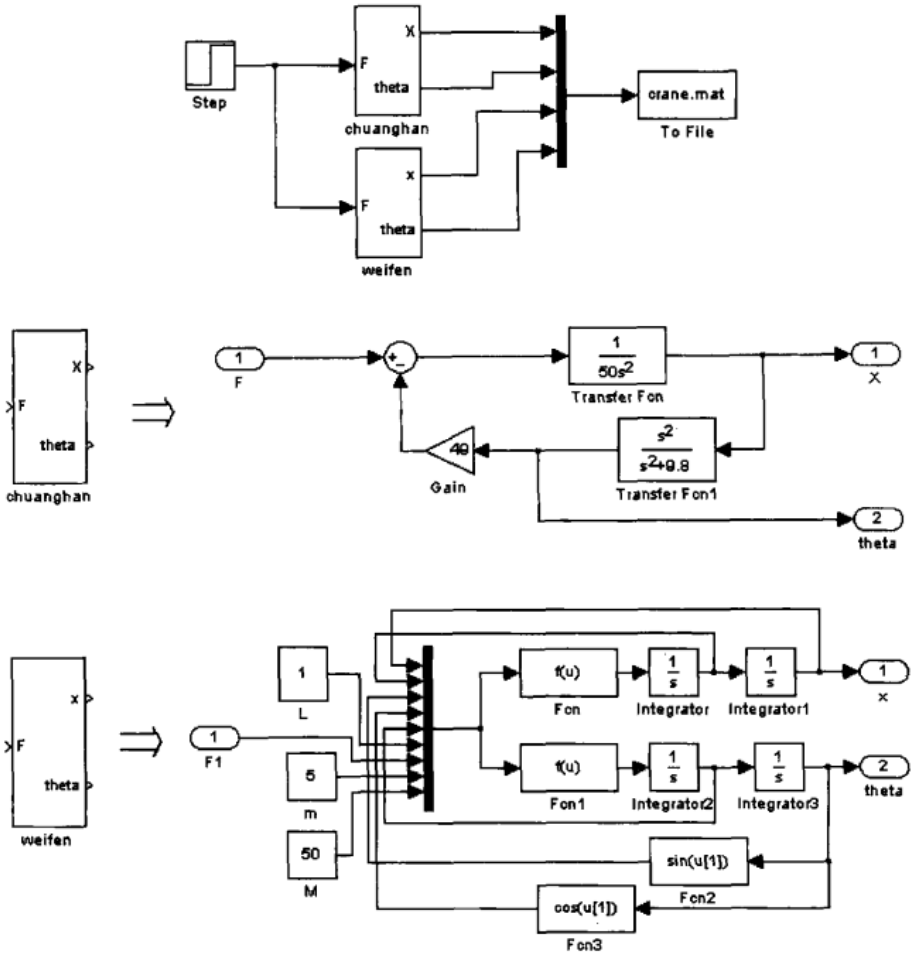


图 4-64　利用子系统封装后的模型框图

所以，在突加恒定作用力拖动下，小车将向前移动，负载将在（$0 \leqslant \theta \leqslant \xi$）区间内摆动（其中 ξ 值与作用力大小有关）。下面利用仿真实验来验证正确数学模型应具有的这一必要性质。

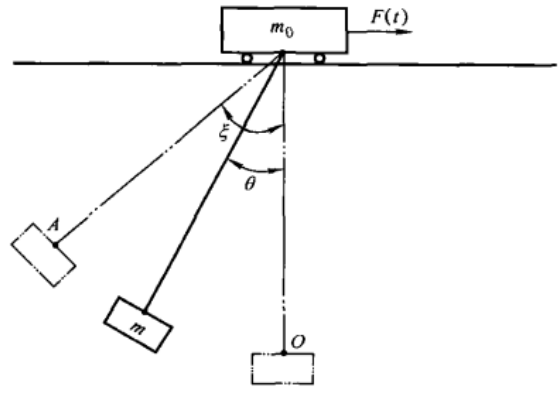


图 4-65　重物摆动原理图

（2）仿真实验　执行图 4-64 所示的程序之结果如图 4-66 所示（该结果曲线绘制时，可通过将第三节中绘制曲线的程序略作修改得到），从中可见，在 1N 恒定拖动力作用下，负载不断地在 $\theta \in [0,\xi]$ 区间内摆动，小

车位移逐渐增加。这一结果符合前述的实验设计，故可以在一定程度上确认：该起重机系统的数学模型是有效的。

同时，由图 4-66 中也可看出，近似模型与精确模型的曲线基本上是重合的，因此，也可以认为近似模型在一定条件下可以表述原系统模型的性质。

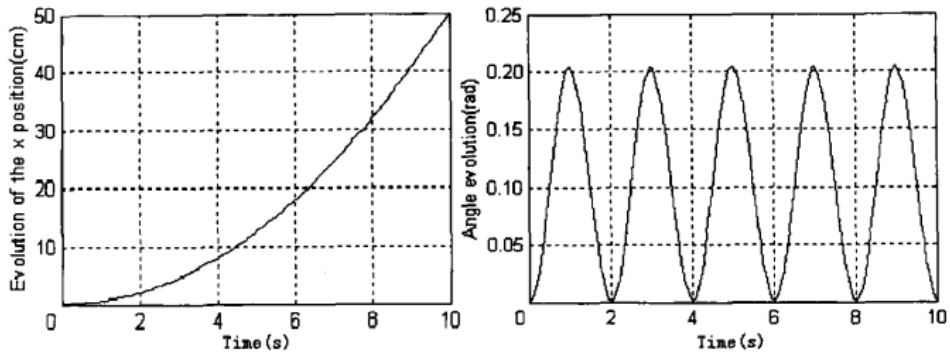


图 4-66 单位阶跃信号时的系统响应曲线

五、时间最优控制策略

由于起重机防摆控制问题的核心是在有效消除重物摆动的条件下，使小车可靠定位，因此，可通过对重物摆动性质的分析，找到负载摆动的特点，以便采取有效的策略进行控制。

（一）负载摆动问题的数学描述

设 $a=\ddot{x}$ 表示小车的水平加速度，则由图 4-62 所示的定摆长起重机系统简化模型可得起重机系统的摆动规律为

$$\ddot{\theta}+\frac{g}{L}\theta=\frac{a}{L}$$

再假设小车的最大加速度为 a_m，设 $u=\dfrac{\ddot{x}}{a_m}$，则有

$$|u|\leqslant 1$$

这样负载摆动方程可写成

$$\ddot{\theta}+\frac{g}{L}\theta=\frac{a_m}{L}u$$

为了便于进行分析，将上式化为状态方程形式，令 $\omega^2=\dfrac{g}{L}$，$K=\dfrac{a_m}{L}$；取状态为 $x_1=\dfrac{\omega}{K}\theta$，$x_2=\dfrac{1}{K}\dot{\theta}$，得到负载摆动的状态方程为

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix}=\begin{bmatrix}0 & \omega\\ -\omega & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}+\begin{bmatrix}0\\ 1\end{bmatrix}u$$

针对该状态方程描述的起重机系统摆动规律，根据系统的控制要求，提出系统控制的性能指标为

$$J=\int_0^{t_f}\mathrm{d}t$$

即要寻求一种最优控制规律 $u(t)$，其能使状态从原点（0，0）（起动前或制动前）经过一个中间状态（ζ_1，ζ_2）后，以最短的时间重新回到原点（摆动为零）。

（二）相平面分析

为研究负载的摆动规律（以实现有效的控制），显然依据相轨迹分析最为直接，因此，我们首先需要在相平面上绘制系统的相轨迹。

负载摆动状态方程的状态转移矩阵为

$$\boldsymbol{\Phi}(t,0)=\mathrm{e}^{At}=\mathrm{e}^{\begin{bmatrix}0 & \omega\\ -\omega & 0\end{bmatrix}t}=\begin{bmatrix}\cos\omega t & \sin\omega t\\ -\sin\omega t & \cos\omega t\end{bmatrix}$$

其解为

$$\begin{bmatrix}x_1(t)\\ x_2(t)\end{bmatrix}=\boldsymbol{\Phi}(t,0)\begin{bmatrix}x_1(0)\\ x_2(0)\end{bmatrix}+\int_0^t\boldsymbol{\Phi}^{-1}(\tau,0)\begin{bmatrix}0\\ u(\tau)\end{bmatrix}\mathrm{d}\tau$$

$$=\begin{bmatrix}\cos\omega t & \sin\omega t\\ -\sin\omega t & \cos\omega t\end{bmatrix}\begin{bmatrix}x_1(0)\\ x_2(0)\end{bmatrix}+\int_0^t\begin{bmatrix}\cos\omega(t-\tau) & -\sin\omega(t-\tau)\\ \sin\omega(t-\tau) & \cos\omega(t-\tau)\end{bmatrix}\begin{bmatrix}0\\ u(\tau)\end{bmatrix}\mathrm{d}\tau$$

为了使时间最优，取控制规律

$$u(t)=\Delta\quad(\Delta=\pm1,0)$$

则可得

$$\begin{cases}x_1(t)=\left[x_1(0)-\dfrac{\Delta}{\omega}\right]\cos\omega t+x_2(0)\sin\omega t+\dfrac{\Delta}{\omega}\\ x_2(t)=-\left[x_1(t)-\dfrac{\Delta}{\omega}\right]\sin\omega t+x_2(0)\cos\omega t\end{cases}$$

设 $\omega x_1(t)$，$\omega x_2(t)$ 为新的状态，上式可整理为

$$[\omega x_1(t)-\Delta]^2+[\omega x_2(t)]^2=[\omega x_1(0)-\Delta]^2+[\omega x_2(0)]^2$$

以 $\omega x_1(t)$ 为横坐标，以 $\omega x_2(t)$ 为纵坐标，可作出系统负载摆动的相平面图如图 4-67 所示。

由图 4-67 可知，系统的相平面轨迹为一个以 $(\Delta,0)$ 为圆心，以 $R=\sqrt{[\omega x_1(0)-\Delta]^2+[\omega x_2(0)]^2}$ 为半径的圆（顺时针运动，旋转的角度为 $\theta=\omega t$）。当处于圆点时（即 $\omega x_1(0)=\omega x_2(0)=0$），摆角与其加速度均为零，即摆动为零；若小车的加速度为零，小车处于匀速运动状态，则不产生摆动。振荡的幅值为 $A(t)=\sqrt{[\omega x_1(t)]^2+[\omega x_2(t)]^2}$，摆动的幅值与状态振荡的幅值相关，即 $A(t)$ 越大，摆动的角度越大。

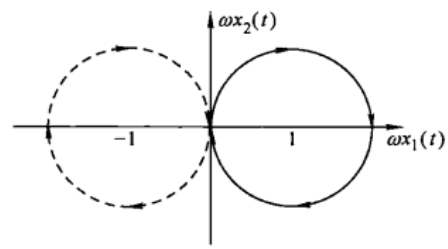


图 4-67　系统相平面

（三）消摆控制策略

1. 消摆策略

由前面的分析可知，当启动时，始终使加速度处于最大，即 $u(t)=1$，摆动的状态运动轨迹如图 4-67 中的右半平面的实线所示，若当状态运动一周回到原点时，摆角及摆速均为零。此时

停止加速，使小车处于匀速运动状态，则摆动将被消除，继续匀速行走时重物将没有摆动现象。同理，当制动停车过程中，始终以最大减速度减速，即 $u(t) = -1$，摆动状态运动轨迹如图 4-67 虚线所示。当状态运动回到原点时，停止减速，则摆动也将被消除，同时，完成负载定位。

该消摆控制策略加减速的时间 t_s 相同，均为走完一个完整圆周使用的时间，即有

$$\omega t_s = 2\pi, \left(\omega^2 = \frac{g}{L}\right)$$

2. 消摆控制策略中的时间切换序列

假设起重机的起动制动最大加速度为 a_m，最大速度为 v_m，绳长为 l，要行走的距离为 s。下面给出消摆的切换时间序列。

当满足 $t_s = 2\pi\sqrt{\frac{l}{g}}$ 时，如图 4-68 所示，可以算出时间切换的各个时刻：

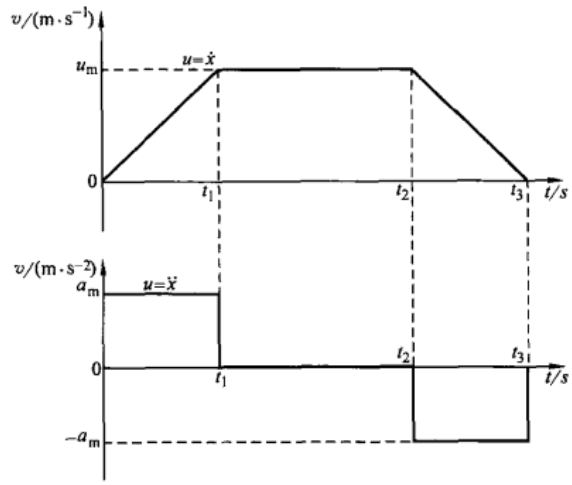


图 4-68 起制动切换时刻

$$t_1 = t_s = 2\pi\sqrt{\frac{l}{g}}$$

$$t_2 = \frac{S}{v_m}$$

$$t_3 = t_1 + t_2 = \frac{S}{v_m} + 2\pi\sqrt{\frac{l}{g}}$$

六、仿真实验

综合上述内容，建立图 4-69 所示的“起重机防摆控制系统” Simulink 仿真系统结构图；其中，控制器输出为加速度，经乘以小车质量后等效为作用在小车上的拖动力（此处有近似）。

其中：

（一）绘图子程序

```
% crane
% singnals recuperation
```

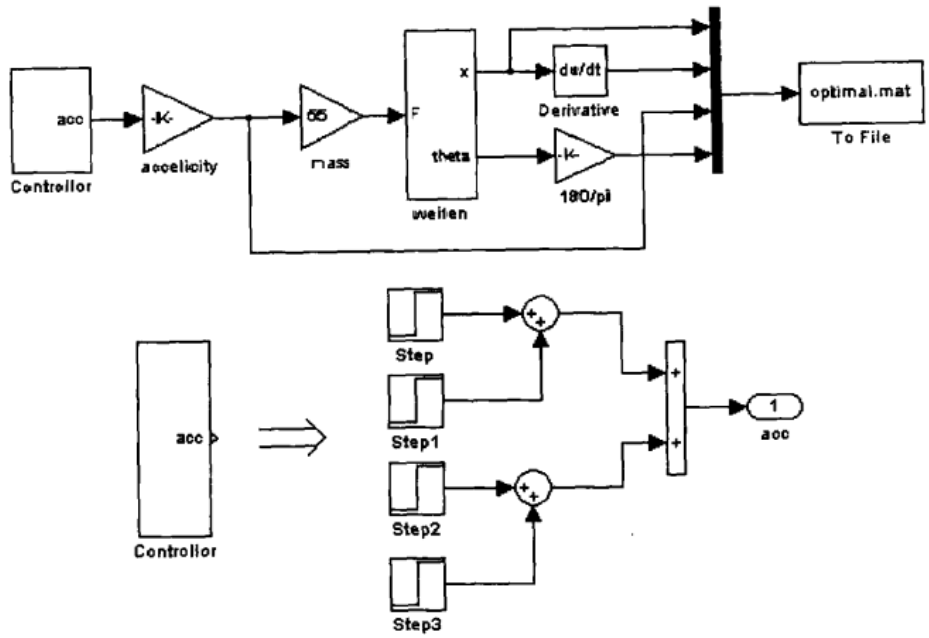


图 4-69　Simulink 仿真框图

```
load optimal. mat
t = signals(1,:);
p = signals(2,:);
v = signals(3,:);
a = signals(4,:);
theta = signals(5,:);
% drawing control and teta(t) response signals
subplot(4,1,1);
plot(t,p);
grid on;
ylabel('positon');
subplot(4,1,2);
plot(t,v);
grid on;
ylabel('velocity')
subplot(4,1,3);
plot(t,a);
grid on;
ylabel('acceleration')
subplot(4,1,4);
plot(t,theta);
```

```
grid on;
ylabel('angle')
xlabel('Time')
```

（二）仿真结果

图 4-70 给出了系统在最大加速度为 $0.5m/s^2$，小车行走距离为 5m，摆长为 1m 条件下的仿真实验结果。从中可见，经“时间最优控制”后的起重机系统，重物的摆动得到有效减小，仿真结果重现了理论分析结果。

因此，基于时间最优控制的消摆控制方案是有效的。

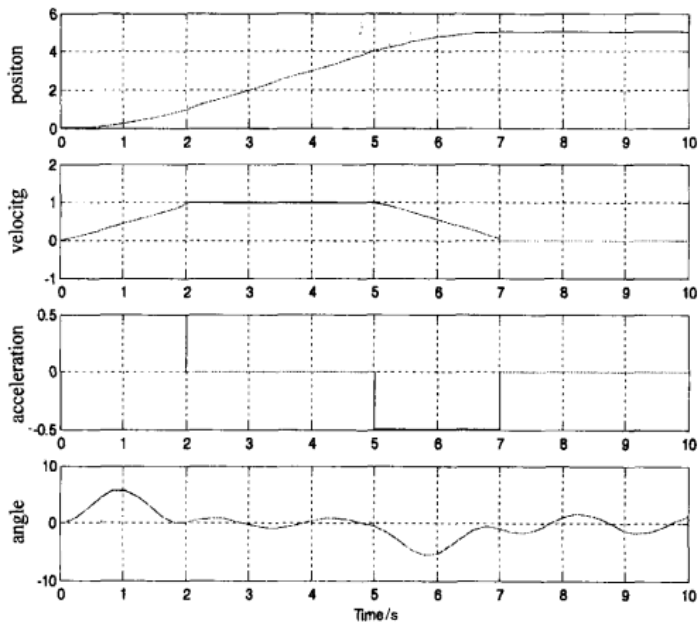


图 4-70 系统仿真结果图

七、结论

1）本节从理论上证明了所设计的基于时间最优控制的起重机防摆控制方案是可行的。

2）本节的结果在实际应用时（实物仿真）还有如下特点：

① 在实际控制过程中不需要检测摆角，从而使得控制系统比较简单。

② 在实际应用过程中该控制方法要求电动机的加速度能够得到很好的控制，同时要求摆角的初始状态必须为零。

③ 由于该时间最优控制方案实质上为开环控制，因此，在实际应用过程中需要自行增加位置环，以保证小车的定位精度。

3）图 4-70 给出的仿真结果中尽管证明了控制策略的有效性，但是其最终还是没有完全消除摆动。这其中有两方面的原因：一是用控制器输出（加速度）乘以小车质量后系“近似等效”为作用在小车上的拖动力（当重物摆动时，作用在小车上的合力是个变量）；二是系统的防摆控制策略设计是在近似模型上进行的，而我们最终的系统仿真实验是在精确模型上进行的。这些近似是造成仿真结果与理论分析间有误差的主要原因，感兴趣的读者可以针对这个问题进一步展开讨论。

4）起重机防摆控制问题也是一个非常典型且具有明确物理意义的运动控制系统问题，同样也可以采用双闭环 PID 控制、模糊控制、线性二次型最优控制等算法进行有效控制，感兴趣的读者可以仿照书中的相关内容进行仿真实验研究。

第六节　问题与探究——“球车系统”的建模与控制问题

一、问题提出

一阶直线倒立摆的物理模型是 20 世纪 50 年代由美国麻省理工学院的控制论专家根据火箭发射助推器的控制问题提出的，现已为大家所熟悉。在过去的半个多世纪里，陆续涌现出了多种新结构的倒立摆，如直线倒立摆（二级倒立摆、多级倒立摆）、环形倒立摆、平面倒立摆、旋转倒立摆等，参考文献［92，93］对它们有详细论述。

球车系统的概念产生于 20 世纪 70 年代，但对其的研究一直停留在仿真层面。美国卡普兰大学的 Ka C. Cheok 和 Nan K. Loh 最早对球车系统的实物设计和控制进行了研究，具体可参见参考文献［94，95］。图 4-71 所示的“球车系统”为 1998 年丹麦技术大学的 Jan Jantzen 制作的实物平台。

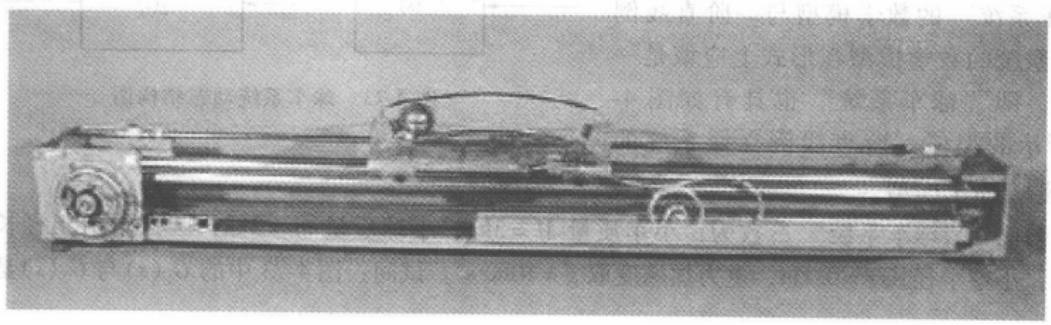


图 4-71　球车系统

“球车系统”的机械结构是由直线滑轨、带有圆弧形滑轨的小车及金属小球三部分组成的。当小车在电动机的控制下在直线滑轨上运动时，金属小球会相对小车上的圆弧形滑轨滚动。系统的控制目标是小车可在直线滑轨上任意移动位置的同时，使金属小球位于圆弧形滑轨的正中央并保持稳定。

对于“球车系统”的控制问题，由于小球在滑轨上位置的测量不能使用传统的旋转电位器或光电编码器，所以对金属小球位置的检测是实物控制系统设计中的一个难点。

二、系统建模

“球车系统”结构如图 4-72 所示。

设小车质量为 M，小球质量为 m，小车位移为 x，圆弧形导轨半径为 R，小球旋转半径为 r，小球转动惯量为 J，作用在小车上的水平方向的力为 F。

利用分析力学中的拉格朗日方程，可推导出系统的动力学模型为

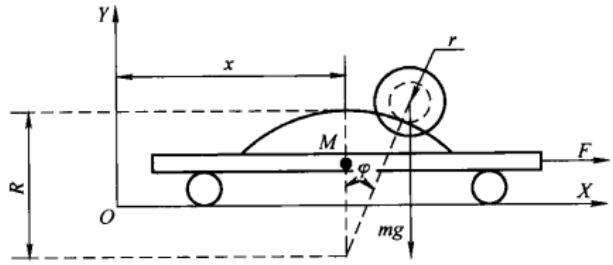


图 4-72 球车系统结构原理图

$$\begin{cases}(M+m)\ddot{x}+m(R+r)\cos\varphi\cdot\ddot{\varphi}-m(R+r)\sin\varphi\cdot\dot{\varphi}^2=F\\ m(R+r)\cos\varphi\ddot{x}+\left[m(R+r)^2+J\left(\dfrac{R+r}{r}\right)^2\right]\ddot{\varphi}-mg(R+r)\sin\varphi=0\end{cases}\tag{4-8}$$

三、问题探究

（一）建模问题

为了便于系统分析与控制器设计，需要将式（4-8）所示的系统精确数学模型进行线性化处理，以得到其线性系统的“传递函数”模型。

有人说，根据“类比分析”原理，“球车系统”的数学模型与一阶直线倒立摆系统的数学模型在形式上应该是一致的，即“球车系统”也具有如图 4-73 所示的类似一阶直线倒立摆系统的“传递函数”模型。

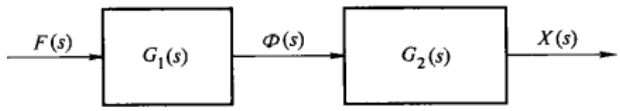


图 4-73 球车系统动态结构图

若已知“球车系统”参数为：小车质量 $M=3\text{kg}$，小球质量 $m=2\text{kg}$，圆弧轨道半径长 $R=0.9\text{m}$，小球半径长 $r=0.1\text{m}$，重力加速度取 $g=10\text{m/s}^2$。试问：图 4-73 中的 $G_1(s)$ 与 $G_2(s)$ 是怎样的？

感兴趣的读者不妨“类比”一阶直线倒立摆系统模型来探究一下。

（二）检测问题

在“球车系统”中，小车位移的测量可以仿照一阶直线倒立摆系统进行，但金属小球在圆弧上位置的测量却是个难题。

感兴趣的读者不妨自己设计一下金属小球位置检测环节。

本节相关内容的进一步了解，感兴趣的读者可登录网址：www. erudit. de/erudit/demos/cartball/index. htm ; www. ifr. mavt. ethz. ch；也可参阅参考文献［95，96］。

小 结

控制系统 CAD 作为一种强有力的工具，极大地促进了自动控制理论的应用与发展。MATLAB 作为当今最具代表性的工具软件环境，为控制系统的建模、分析与设计创造了一个良好的工作平台，是我们应重点掌握的内容。如何灵活、高效地运用已有的 CAD 软件系统，是我们今后要面临的重要问题，只有勤于思考、善于总结、努力实践，才能使我们真正体会

到控制系统数字仿真与 CAD 技术给我们所带来的巨大效益和乐趣。

习　题

4-1　设控制系统的开环传递函数为

$$G(s)H(s) = \frac{K(s+1)}{s(s-1)(s^2+4s+16)}$$

试画出该系统的根轨迹。

4-2　某反馈控制系统的开环传递函数为

$$G(s)H(s) = \frac{K}{s(s+4)(s^2+4s+20)}$$

试绘制其根轨迹。

4-3　已知某系统传递函数为

$$W(s) = \frac{80\left(\frac{1}{100}s+1\right)}{\left(\frac{1}{40}s+1\right)\left[\left(\frac{s}{200}\right)^2+2\times 0.3\times\frac{1}{200}s+1\right]}$$

试绘制其博德图。

4-4　设控制系统具有如下的开环传递函数

$$G(s)H(s) = \frac{K}{s(s+1)(s+5)}$$

试求取当 $K=10$ 时的相角裕度和幅值裕度，并画出其博德图。

4-5　已知某单位反馈系统开环传递函数为

$$G(s) = \frac{1}{s(0.1s+1)(0.02s+1)(0.01s+1)(0.005s+1)}$$

若性能指标要求如下：$\gamma=45°$，$K_v=200$，$\omega_c=13.5$，试确定校正装置。

4-6　某过程控制系统如图 4-74 所示，试设计 PID 调节器参数，使该系统动态性能达到最佳。

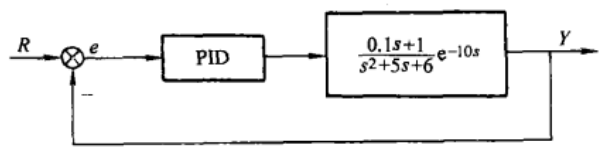


图 4-74　题 4-6 图

4-7　试采用 smith 预估控制方法对题 4-6 所述系统进行重新设计，并用仿真的方法分析滞后参数变化对系统动态性能的影响。

4-8　图 4-75 所示为一带有库仑摩擦的二阶随动系统，试优化设计 K_1 参数，并分析非线性环节对系统动态性能的影响。

4-9　试分析图 4-76 所示系统中死区非线性对系统动态性能的影响。

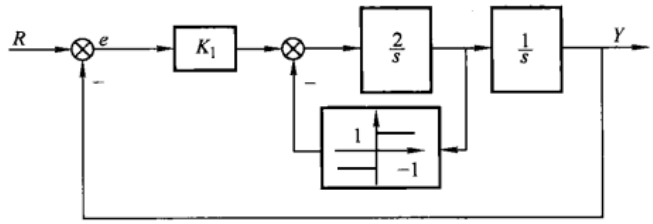


图 4-75 题 4-8 图

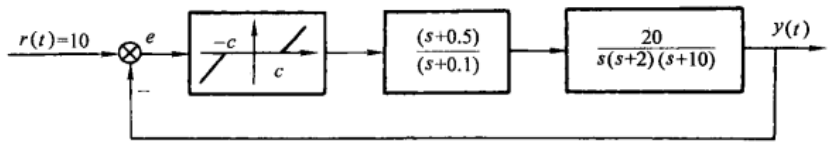


图 4-76 题 4-9 图

4-10 图 4-77 所示为计算机控制系统，试设计一最小拍控制器 $D(z)$，并用仿真的方法分析最小拍控制器对系统输入信号和对象参数变化的适应性。

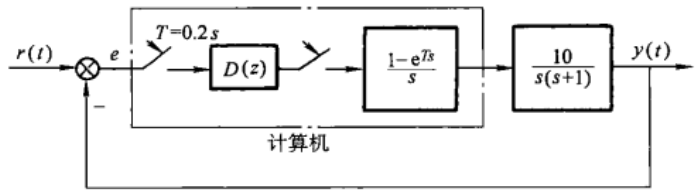


图 4-77 题 4-10 图

4-11 为使图 4-78 所示系统不产生自激振荡，试分析 a、b 取值。

4-12 已知某一地区在有病菌传染下的描述三种类型人数变化的动态模型为

$$\begin{cases} \dot{X}_1 = -\alpha X_1 X_2 & X_1(0) = 620 \\ \dot{X}_2 = \alpha X_1 X_2 - \beta X_2 & X_2(0) = 10 \\ \dot{X}_3 = \beta X_2 & X_3(0) = 70 \end{cases}$$

式中，X_1 表示可能传染的人数；X_2 表示已经得病的人数；X_3 表示已经治愈的人数；$\alpha = 0.001$；$\beta = 0.072$。

试用仿真方法求未来 20 年内三种人人数的动态变化情况。

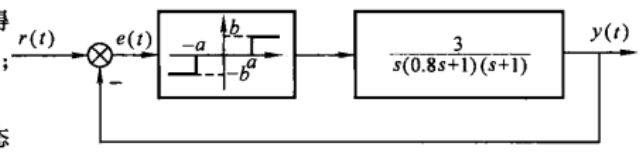


图 4-78 题 4-11 图

4-13 对于高阶系统的设计问题，往往都要进行降阶近似处理，并要验证近似效果。已知某高阶系统模型为

$$G(s) = \frac{194480 + 422964s + 511812s^2 + 278376s^3 + 82402s^4 + 13285s^5 + 10861s^6 + 35s^7}{9600 + 28880s + 37492s^2 + 27470s^3 + 11870s^4 + 3017s^5 + 437s^6 + 33s^7 + s^8}$$

经简化处理后，模型等效为

$$G(s) = \frac{284.98 + 35s}{12.31 + 10.114s + s^2}$$

试比较两个模型在单位阶跃信号作用下的响应情况，并分析近似效果。

4-14 针对如图 4-79 所示的一阶双摆系统，试讨论如下问题（两均匀杆所用材料相同）：

1）控制量 F（施加的作用力）能否在保持双摆不倒的前提下，实现小车的位置伺服控制？

2）试给出你的具体实现方案。（提示：说明系统的实物制作原理与控制策略）

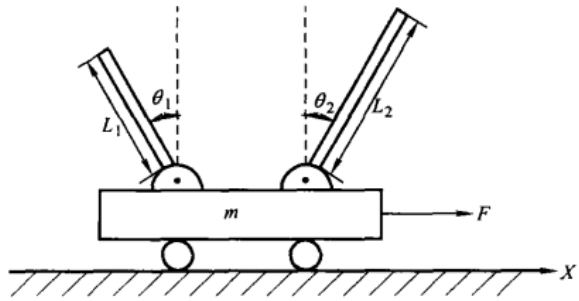


图 4-79　题 4-14 一阶双摆系统原理图

4-15　图 4-80 所示为基于视觉传感器的磁悬浮轴承运动控制系统，它将磁悬浮轴承的控制问题抽象成利用电磁铁将一个金属轴杆悬挂起来的问题。此系统能够利用摄像头采集的图像计算出轴杆与电磁铁之间的距离 x，然后由计算机计算出为实现平衡所需要的控制电流，最后通过驱动电路对驱动电流实施控制。

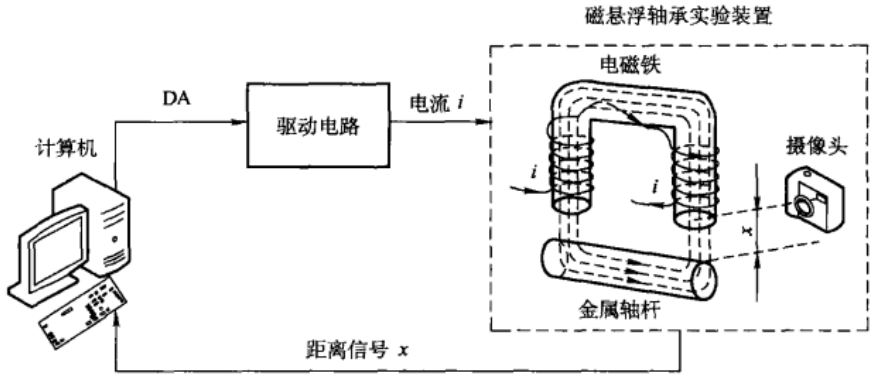


图 4-80　题 4-15 图

假设令 i_0 表示平衡点的工作电流，则本运动控制系统的微分方程可以在平衡点处线性化为

$$m\ddot{x} = k_1 x + k_2 i$$

一组合理的参数值为 $m = 0.02\text{kg}$，$k_1 = 20\text{N/m}$，$k_2 = 0.4\text{N/A}$。

试解答下述问题：

1）用极点配置法为此磁悬浮轴承控制系统设计一个控制器，使闭环系统满足如下指标：调整时间 $t_s \leqslant 0.25\text{s}$，对 x 的某一初始偏移的超调量小于 20%。

2）为你设计的系统绘制关于 k_1 的根轨迹，并讨论能否用你设计的闭环系统平衡各种质量的轴杆。

3）假定一个初始阶跃位移扰动作用于轴杆，且传感器仅能测量距离工作点 $\pm 0.25\text{cm}$ 范围内的 x，驱动电路仅能提供 1A 的控制电流，那么可能控制的最大位移是多少？

第五章　数字仿真技术的综合应用

在仿真技术的应用中涉及许多相关内容，本节将以 MATLAB 为基础就几个方面的内容进行介绍，着重阐述电动机速度控制系统设计与分析、数字控制器设计与应用以及典型控制系统的建模与仿真分析等问题。有关问题所涉及的基础内容，鉴于篇幅所限不能详尽阐明，请读者参考有关专著。

第一节　直流电动机双闭环调速系统设计中的若干问题

自 20 世纪 70 年代以来，国内外在电气传动领域里，大量地采用了“晶闸管整流电动机调速”技术（简称 V-M 调速系统）。尽管当今功率半导体变流技术已有了突飞猛进的发展，但在工业生产中 V-M 调速系统的应用量还是占有相当比重的。一般情况下 V-M 调速系统均设计成图 5-1 所示的转速、电流双闭环形式。

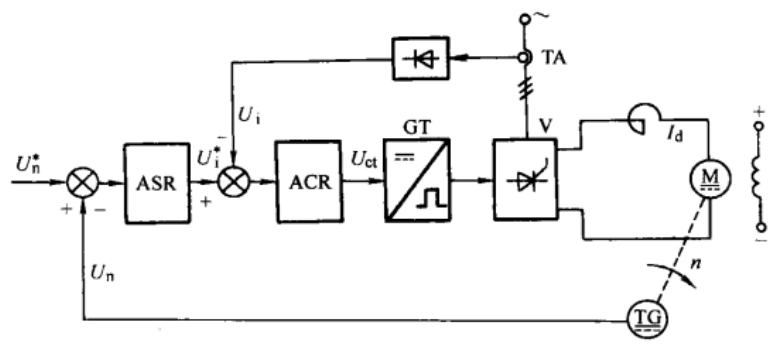


图 5-1　直流电动机双闭环调速系统结构图

一、双闭环 V-M 调速系统的目的

图 5-1 给出的是一双闭环直流调速系统，它着重解决了如下两方面的问题：

（一）起动的快速性问题

借助于 PI 调节器的饱合非线性特性，使得系统在电动机允许的过载能力下尽可能地快速起动。理想的电动机起动特性如图 5-2 所示。

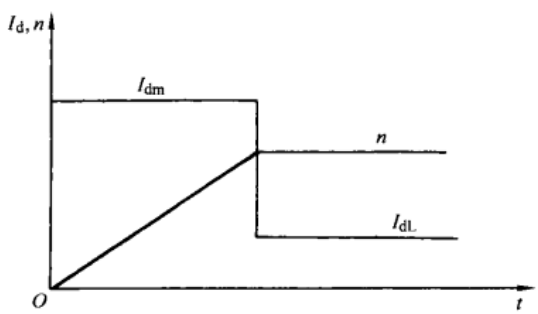


图 5-2　理想电动机起动特性

（二）提高系统抗扰性能

通过调节器的适当设计可使系统转速对于电网电压及负载转矩的波动或突变等扰动予以迅速抑制，在恢复时间上达到最佳，其动态特性如图 5-3 所示。

二、关于积分调节器的饱和非线性问题

双闭环 V-M 调速系统中的 ASR 与 ACR 一般均采用 PI 调节器，其中有积分作用（I 调节）。在图 5-4 中给出了控制系统的 PI 控制规律及系统简要结构，从中我们可知：

1）只要偏差 $e(t)$ 存在，调节器的输出控制电压 U 就会不断地无限制地增加（正向或负向）。因此，必须在 PI 调节器输出端加限制装置（即限幅 U_m）。双环 V-M 调速系统 ASR 的限幅输出为 U_{im}^*，对应于最大电流给定。

2）当 $e(t)=0$ 时，$U=$ 常数。若要使 U 下降，必须使 $e(t)<0$。因此，在调速系统中若要使 ASR 退出饱和输出控制状态，一定要有超调产生。

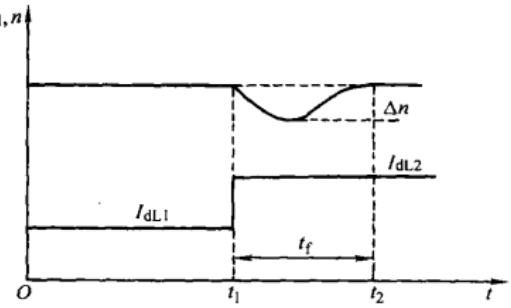


图 5-3　双环系统负载扰动特性

3）若控制系统中（前向通道上）存在有积分作用的环节（调节器，对象），则在给定作用下，系统输出一定会出现超调。

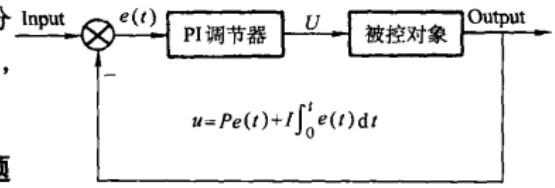


图 5-4　具有积分控制作用的系统结构

三、关于 ASR 与 ACR 的工程设计问题

对于电动机调速控制系统，通常调节器均按 PI 形式设计，将对象综合成典型系统，如图 5-5 所示。

对于双闭环 V-M 调速系统，电流环通常按典型 I 型系统设计，而转速环按典型 II 型系统设计，这是因为 I 型系统跟随性能好一些，而 II 型系统抗扰性能略佳。作为图 5-1 所示双闭环控制系统 PI 调节器的参数设计，在理论上有如下电子最佳设计方法。

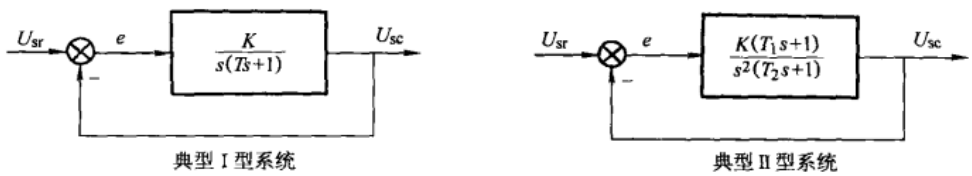


图 5-5　典型系统结构图

1）电流调节器：$D(s)=K_i\dfrac{1+\tau_i s}{\tau_i s}$

取　　$\tau_i=T_l$

$$K_I=\frac{1}{2T_{\Sigma i}},\quad (T_{\Sigma i}=T_s+T_{oi})$$

$$K_i=K_I\frac{\tau_i R}{\beta K_s}$$

2）转速调节器：$D(s)=K_n\dfrac{1+\tau_n s}{\tau_n}$

取　　$\tau_n=5\times T_{\Sigma n}\quad (T_{\Sigma n}=2T_{\Sigma i}+T_{on})$

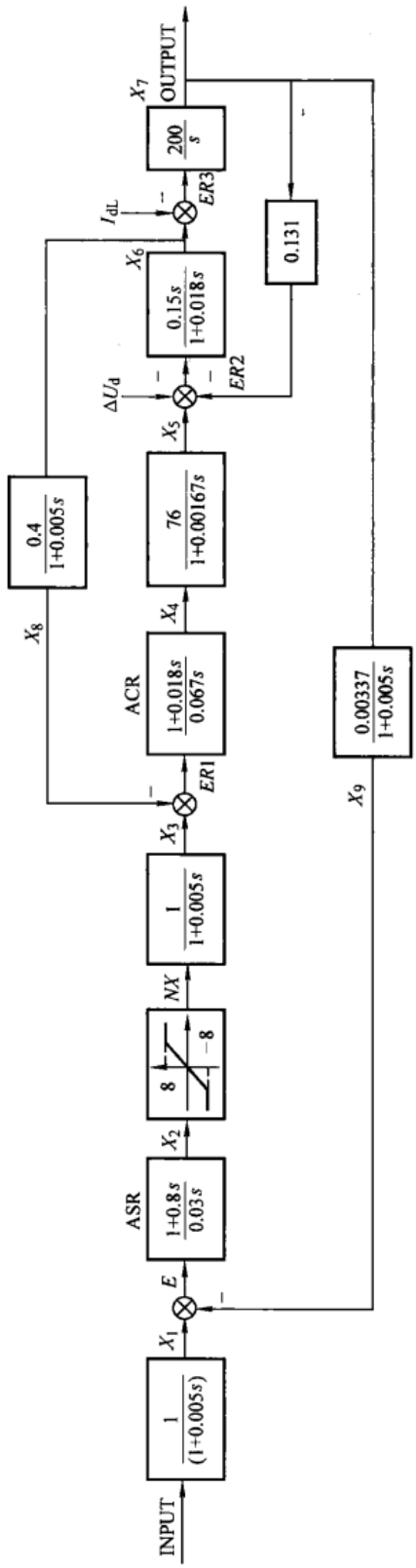


图 5-6 双闭环调速系统动态结构图

$$K_N = \frac{6}{50 \times T_{\Sigma n}^2}$$

$$K_n = K_N \frac{\tau_n \beta C_e T_m}{\alpha R}$$

经过如上设计的 V-M 调速系统从理论上讲有如下动态性能：电动机起动过程中电流的超调量为 4.3%，转速的超调量为 8.3%。

但是，在工程实际中如上的理论设计与实际调试出的参数尚有一定的差距，一般情况下大约有 3～5 倍之多，这一点从下面的仿真分析中也可以看到。

四、双闭环 V-M 调速系统的动态分析

这里我们借助于 SIMULINK 来分析一下双闭环 V-M 调速系统的动态性能。首先，我们应对图 5-1 所示系统建立相应的数学模型。图 5-6 给出了某 V-M 调速系统的动态结构图，其中 ASR 与 ACR 的参数在理论设计的基础上已作适当调整。

系统中采用三相桥式晶闸管整流装置，基本参数如下：

直流电动机：220V，13.6A，1480r/min，$C_e = 0.131\text{V/}(\text{r} \cdot \text{min}^{-1})$，允许过载倍数 $\lambda = 1.5$。

晶闸管装置：$K_s = 76$。

电枢回路总电阻：$R = 6.58\Omega$。

时间常数：$T_l = 0.018\text{s}$，$T_m = 0.25\text{s}$。

反馈系数：

$\alpha = 0.00337\text{V/}(\text{r} \cdot \text{min}^{-1})$，

$\beta = 0.4\text{V/A}$。

反馈滤波时间常数：

$\tau_{oi} = 0.005\text{s}$，$\tau_{on} = 0.005\text{s}$。

（一）起动特性分析

在前面图 5-2 中已给出电动机的理想起动特性，此时电动机的起动速度达到最快。因此，系统设计结果应尽可能接近这一理想特性。在下面图 5-7 的仿真结果中分别给出了该系统 ASR 的输出与电动机转速，ACR 的输出与电动机转速及电动机电流与电动机

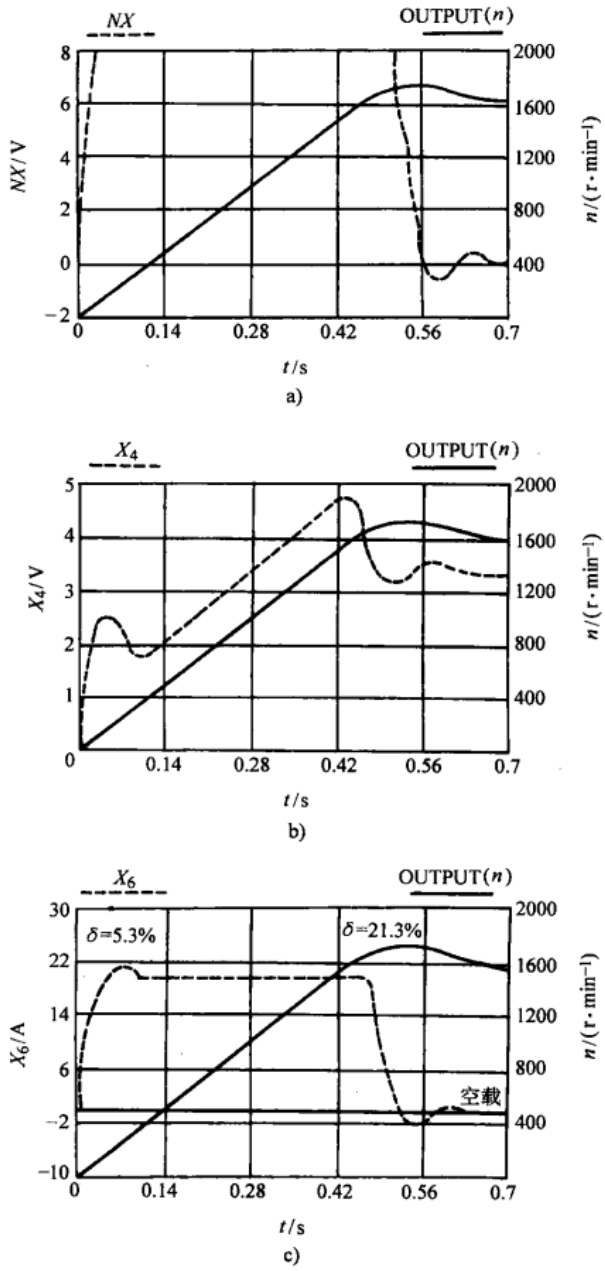


图 5-7　双闭环 V-M 调速系统的起动特性

a）ASR 的输出特性　b）ACR 的输出特性

c）电动机电流特性

转速在起动过程中的动态特性仿真结果。

通过如上的仿真分析，对于图 5-6 所示系统的工作过程可概括成如下几点：

1）ASR 从起动到稳速运行的过程中经历了两个状态，即饱和限幅输出与线性调节状态。

2）ACR 从起动到稳态运行的过程中只工作在一种状态，即线性调节状态。

3）图 5-7 所示的电动机起动特性已十分接近理想特性。所以，该系统设计对于起动特性来说，已达到预期目的。

4）对于系统性能指标来说，起动过程中电流的超调量为 5.3%（δ = 5.3%），转速的超调量达 21.3%（δ = 21.3%）。显然这一指标与理论最佳设计尚有一定的差距，尤其是转速的超调量略高一些。

（二）抗扰性能分析

一般情况下，双闭环调速系统的干扰主要是负载突变与电网电压波动这两种。因此在图 5-8 中分别给出了该系统电动机转速在突加负载及电网电压突减情况下动态特性的仿真结果。

通过如上的仿真分析，对于该系统的抗扰性能来说我们可有如下几点结论：

1）系统对负载的大幅度突变具有良好的抵抗能力，在 $\Delta I = 12A$ 的情况下系统速降为 $\Delta n = 44r/min$，恢复时间 $t_f = 1.5s$。

2）系统对电网电压的大幅度波动同样也具有良好的抵抗能力。在 $\Delta U = 100V$ 的情况下，系统速降仅为 9r/min，恢复时间 $t_f = 1.5s$。

3）与理想的电动机的起动特性相比较，该系统的起动及恢复时间显得略长一些（轻载状态下接近 4s）。

（三）存在问题分析

1）仿真结果与理论设计具有一定的差距，由图 5-8 可见，电流动态响应的超调量为 $\delta = 5.3\%$（理论值为 4.3%），转速动态响应超调量为 $\delta = 21.3\%$（理论值为 8.3%）。另外 ASR 与 ACR 的参数与理论设计值也有差距。

结合实际情况来分析，不难理解如上所述的差距。由于前面所讲的“典型系统的电子最佳设计方法”考虑了理论分析与设计的简便性，所以对“双闭环 V-M 调速系统”做了一些简化处理。这里的简化主要是将非线性问题做线性化处理。如滞后环节近似为一阶惯性，调节器的输出限幅特性近似为线性环节等。

以上所谈的近似问题在实际仿真中并

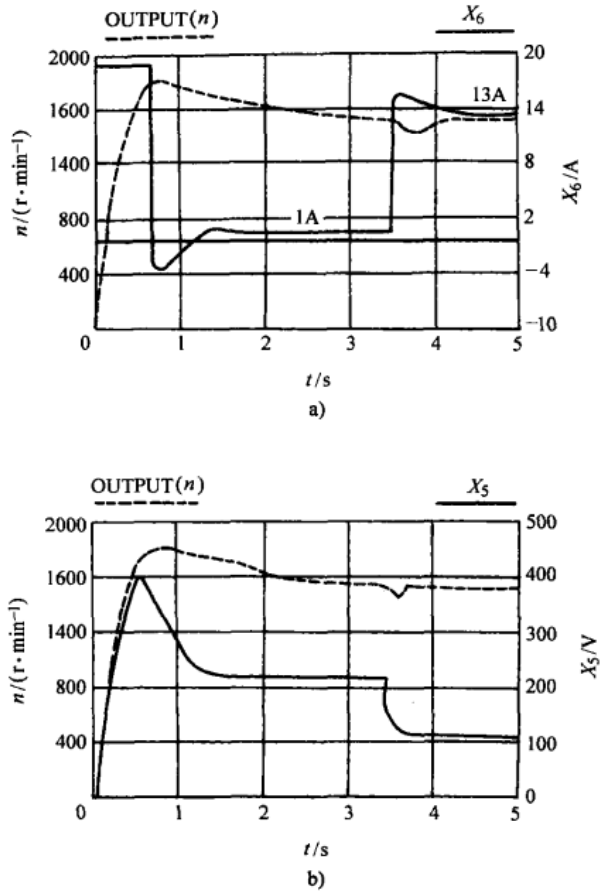


图 5-8 双闭环 V-M 调速系统的抗扰特性

a）突加负载抗扰特性 b）电网电压突减抗扰特性

没有完全采用，而是从实际情况出发等效的，如 ASR 的输出就引入了饱和非线性环节。正是由于理论分析的近似与仿真实验的不完全近似才带来了结果上的差距。

2）从仿真结果上看，系统还未调整到最佳状态，这一点可从动态过渡过程时间及动态恢复时间上来看，对于小功率电动机来说系统的响应慢了一点。

如果将 ASR 与 ACR 的参数作进一步调整的话（或引入补偿控制）还可使系统动态参数进一步改善，当然这需要经验与时间，感兴趣者不妨一试。

3）如果在实际装置上再对系统进行实际调整，我们还会发现：实际调试结果与仿真结果还将有一些差距，而与理论分析结果的差距可能更大一些（主要的调节器参数）。

这一现象可从两方面加以分析：一方面是仿真模型（主要是电动机模型）的不确定性问题；另一方面是系统模型的参数时变问题，如整流装置的内阻 R_{rec}。总的来说，仿真所用系统模型的不准确及系统模型参数本身的不确定性是产生仿真结果与实际调试结果之间存在差距的主要原因。

第二节　数字 PID 调节器的鲁棒性设计方法

在当今的自动控制系统中，由于 PID 调节器具有结构简单、性能稳定可靠等优点，广泛为人们所采用。但是，PID 控制在实际应用中仍存在一定的问题，归结起来主要表现为“参数整定困难”。

对于具有某种不确定性的控制系统，传统的 PID 参数整定方法很难使控制效果达到理想境地。因此，人们总是试图在“参数自适应”及“智能化控制”等方面予以推广。而对于确定性系统的 PID 调节器参数的设计又存在着“方法繁杂、应用不便”等问题。

1985 年，匈牙利国家科学院 Cs. Banyasz 等人提出了一种“数字 PID 调节器的鲁棒性设计方法”[32]，由于它是以“使系统稳定裕度最佳”为原则，具有算式简单、易于在数字计算机上应用以实现自适应控制等优点，广为工程控制界所瞩目。下面我们在简要介绍其基本原理的基础上给出该方法在“高精度齿轮量仪”电控系统上的应用结果，从而证明该方法的可行性。

一、数字 PID 调节器的鲁棒性设计

一般情况下，系统控制对象总可以用如下线性差分方程形式来描述

$$A(z^{-1})y(k) = z^{-d}B(z^{-1})u(k) + C(z^{-1})\varepsilon(k) \tag{5-1}$$

也称之为 CARMA 模型。式中

$$A(z^{-1}) = 1 + a_1z^{-1} + a_2z^{-2} + \cdots + a_{n_a}z^{-n_a}$$

$$B(z^{-1}) = b_0 + b_1z^{-2} + b_2z^{-2} + \cdots + b_{n_b}z^{-n_b}$$

$$C(z^{-1}) = 1 + c_1z^{-1} + c_2z^{-2} + \cdots + c_{n_c}z^{-n_c}$$

$y(k)$ 为可测输出量；$u(k)$ 为可测输入量；$\varepsilon(k)$ 为不可测扰动量，且 $E[\varepsilon(k),\varepsilon(k)\varepsilon^T(k)] = E(0,1)$。

通过对系统模型参数进行辨识，总可以确定模型参数 n_a、n_b、n_c 及 a_i、b_i、c_i。根据以上所述的对象模型，当不考虑随机干扰的影响时，设计的控制器有如下结构形式

$$\begin{cases} D(k) = \dfrac{u(k)}{e(k)} = \dfrac{1}{(1-z^{-1})}\dfrac{G(z^{-1})}{F(z^{-1})} \\ e(k) = R(k) - y(k) \end{cases} \tag{5-2}$$

其中

$$G(z^{-1}) = g_0 + g_1 z^{-1} + g_2 z^{-2} + \cdots + g_{n_g} z^{-n_g}$$

$$F(z^{-1}) = 1 + f_1 z^{-1} + f_2 z^{-2} + \cdots + f_{n_f} z^{-n_f}$$

通过以上的设计可知，若控制器参数多项式 $G(z^{-1})$、$F(z^{-1})$ 与对象模型中多项式 $A(z^{-1})$、$B(z^{-1})$ 完全对消，则控制系统结构将会大为简化，从而便于对系统进行调整。下面分两种情况进行讨论。

1）当对象为开环稳定的最小相位系统时，设

$$\begin{cases} G(z^{-1}) = g_0(1 + g'_1 z^{-1} + g'_2 z^{-2} + \cdots + g'_{n_g} z^{-n_g}) = g_0 A(z^{-1}) \\ F(z^{-1}) = 1 + f'_1 z^{-1} + f'_2 z^{-2} + \cdots + f'_{n_f} z^{-n_f} = B(z^{-1})/b_0 \\ n_g = n_a, n_f = n_b \end{cases}$$

则系统闭环传递函数简化为

$$y(k) = \frac{g_0 b_0}{z^d - z^{d-1} + g_0 b_0} R(k) \tag{5-3}$$

对于一步滞后系统（即 $d = 1$），由双线性变换可得闭环系统的稳定条件为

$$0 < g_0 b_0 < 2 \tag{5-4}$$

当取 $g_0 b_0 = 1$ 时式（5-3）进一步简化为

$$y(k) = z^{-1} R(k) = R(k-1) \tag{5-5}$$

通过以上分析得如下结论：

对于开环稳定的最小相位系统，伺服系统控制器可设计成零极点对消的形式，通过控制器参数的适当选取 $\left(\text{如 } g_0 = \dfrac{1}{b_0}\right)$，可使系统具有最小拍的跟踪特性。

2）当对象为非最小相位系统时，不能进行零极点的对消；这里我们仅考虑 $n_g = n_a = 2$，$n_b = 1$ 的情况。设

$$G(z^{-1}) = g_0(1 + g'_1 z^{-1} + g'_2 z^{-2}) = g_0 A(z^{-1})$$

$$F(z^{-1}) = 1$$

则系统闭环传递函数简化成如下形式

$$y(k) = \frac{g_0 b_0 (1 + r z^{-1}) z^{-d}}{1 - z^{-1} + g_0 b_0 z^{-d} + g_0 b_0 r z^{-(d+1)}} E(k) \tag{5-6}$$

式中，$r = b_1/b$。

由式（5-6）可见，闭环系统的极点分布与 $g_0 b_0$，r 有关；d 值的大小取决于被控对象的滞后步数，对于伺服系统一般有 $d = 1$。

对于式（5-6）所示的闭环系统，设 $k = g_0 b_0$，选择 k 值，以使系统闭环传递函数的范数为 1，从而有

$$k = \sqrt{\frac{2(1-\cos x)}{1+2r\cos x+r^2}} \tag{5-7}$$

式中，$x = T\omega_0$，T 为采样周期，ω_0 为自振角频率。再选择 x 值，以使超前相位角为 60°（$\pi/3$），将其近似看做为 1，即有非线性方程

$$x = \frac{1-2\arctan[r\sin x/(1+r\cos x)]}{2d-1} \xlongequal{\text{def}} g(x) \tag{5-8}$$

根据式（5-7）、式（5-8），有如下控制器参数设计的迭代法：

1）取迭代方程为

$$x_{i+1} = \frac{1}{2}[x_i + g(x_i)] \tag{5-9}$$

由已知的 r 值及假设的初值 x_0，据式(5-8) 可得 $g(x_0)$，将其代入式(5-9) 可得 x_1，如此反复迭代，直至满足我们所需的精度为止，一般地，当 $|x_{i+1}-x_i| < \varepsilon$ 时，即停止迭代计算。其中，ε 值是精度要求，为预先给定的小数。

2）由已知 r 值及迭代所得 x 值，由式（5-7）可得 k 值。

3）由 $k = g_0 b_0$ 可解得 g_0，则由假设条件知 $g_1 = g_0 a_1$，$g_2 = g_0 a_2$。

至此，伺服系统控制器的设计就完成了，该方法也称为经验的鲁棒性设计法。一般情况下可使伺服系统具有很小的超调量及较短过渡过程等优良性能。

二、“高精度齿轮量仪”位置伺服系统控制器设计

“高精度齿轮量仪”电控系统的核心是一双轴联动的位置伺服系统，其中主轴电动机直接驱动被测齿轮这一负载。根据“定位测量”的工艺要求，在一个被测齿面上大约要对 2000 个点进行采样处理，而且要反复测量 10 次，以考查其重复测量精度。因此，每一个测量点的定位时间直接影响到整个测量过程的效率。再者，由于被测齿轮的直径变化较大（50～700mm），即负载的惯量变化较大，所以不同的被测齿轮将使得控制系统对象的参数（主要是机电时间常数）发生变化。

鉴于上述原因，要求电控系统的控制器具有使系统动态响应快（无振荡）、鲁棒性好（对系统参数变化不敏感）等特点，而上节所谈的 PID 调节器的设计方法很适于本伺服系统，因此，有必要先对其进行仿真分析，以确定其可行性。

对于所设计的位置伺服系统有如下传递函数形式的对象数学模型

$$G(s) = \frac{K}{s(T_m s+1)} \tag{5-10}$$

其中，$K = 175$，$T_m = 15\text{ms}$。其脉冲传递函数为

$$\begin{aligned} W(z) &= z\left[\frac{1-e^{-T\cdot s}}{s}\frac{175}{s(0.015s+1)}\right] \\ &= \frac{0.0492(1+0.935z^{-1})}{1-1.8187z^{-1}+0.8187z^{-2}}z^{-1} \end{aligned} \tag{5-11}$$

可见，控制对象为二阶、一步滞后系统。

对于最小相位系统，由上节分析结果可得控制器结构为

$$D(z) = \frac{20.33(1-1.8187z^{-1}+0.8187z^{-2})}{(1-z^{-1})(1+0.935z^{-1})} \tag{5-12}$$

即

$$u(k) = u(k-1) + 20.33e(k) - 36.97e(k-1) + 16.644e(k-2) + 0.935[u(k-2) - u(k-1)] \tag{5-13}$$

对于非最小相位系统，由上节分析结果及对象参数可得控制器结构为

$$D(z) = \frac{g_0(1 + g_0'z^{-1} + g_2'z^{-2})}{(1-z)^{-1}} \tag{5-14}$$

其中，$g_1' = a_1 = 1.8187, g_2' = a_2 = 0.8187$。由于 $b_0 = 0.0492, b_1 = 0.046$，所以 $r = 0.935$。设 $\varepsilon = 0.001$，令 $x_1 = 0.5$，由迭代法得

$$x_1 = 0.5 \quad x_2 = 0.5086 \quad x_3 = 0.5087 \quad x_4 = 0.50873$$

可见经四步迭代即满足精度要求。将 x_4 代入式（5-7）得 $k = 0.2687$。所以 $g_0 = k/b_0 = 5.4614$，$g_1 = g_0a_1 = -9.9326, g_2 = g_0a_2 = 4.4712$。则式（5-14）有如下形式的差分方程

$$u(k) = u(k-1) + 5.4614e(k) - 9.9326e(k-1) + 4.4712e(k-2) \tag{5-15}$$

根据以上参数，我们采用 Simulink 进行仿真研究。其结果如图 5-9 所示，其中图 5-9a 为按最小相位系统设计的仿真结果，图 5-9b 为按非最小相位系统设计的仿真结果。

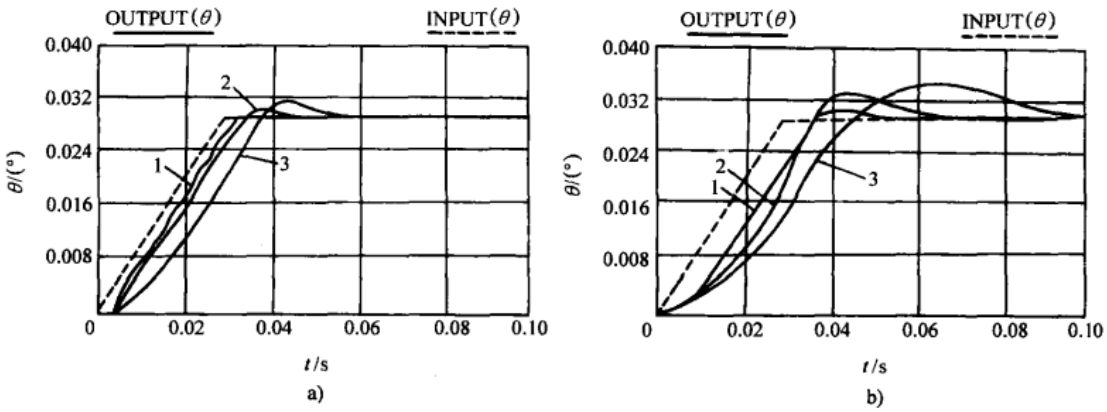


图 5-9 仿真结果

在图 5-9 中，曲线 1 为参数设计准确时的情况，曲线 2、3 为当对象参数（T_m）增加 2 倍和 4 倍时的情况。通过比较可见，就适应系统对象参数变化的能力而言（鲁棒性），前者（图 5-9a）比后者（图 5-9b）要强一些。同传统的 PID 算法相比，式（5-13）所示的控制算法中增加了 $u(k-2)$ 一项，即当前控制的输出要受到“以往”控制输出量的加权约束，其结果使得系统具有某种优良的性能。

从以上的仿真结果可得如下结论：对于模型参数已知的位置伺服系统，所设计的控制器是满足系统要求的，而且具有良好的定位控制性能。

高性能的控制系统需要较为完善的控制方法与手段。本节从实际出发给出了具有良好性能的控制算法，并借助于系统仿真实验为以后的实际工作提供了先验知识。

第三节　“水箱系统”液位控制的仿真研究

在第一章中曾谈到，控制系统的数字仿真实验包括三个基本活动，即模型建立、仿真实验与结果分析。下面，以“水箱系统”液位控制为例逐步完成仿真实验的三个基本活动。

在化工及工业锅炉自动控制系统中，有许多问题最终都可归结为“水箱系统”的液位控制问题，图 5-10 给出了其工艺过程原理。

图中，入口处的阀门由一个调节器控制，以保持水位不变，出口处的阀门由外部操纵，可将其看成一个扰动量。下面用仿真实验法设计一个适合的调节器并确定调节器的参数。

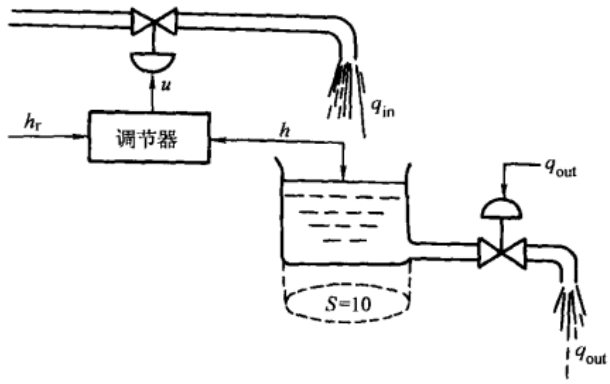


图 5-10　“水箱系统”液位控制原理

一、系统建模

对图 5-10 所示系统可抽象成图 5-11 所示的数学模型。

图中，取

$$K_1 = 1, K_2 = \frac{1}{s} = 0.1, g = 9.81$$

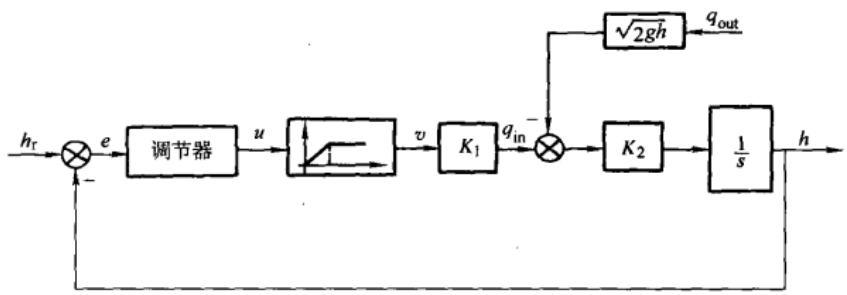


图 5-11　“水箱系统”的数学模型

二、数字仿真

对图 5-11 所示模型，采用 Simulink 进行仿真实验，其中外部干扰信号在 $t = 100$s 时由 0.01 增加到 0.05，调节器采用 PID 算法，当分别采取 P 及 PI 调节规律时，有图 5-12 所示的仿真结果。

三、结果分析

由图 5-12 所示的仿真结果可见：当采用比例调节规律时，系统液位控制存在稳态误差且其随干扰量的增加而增大，而采用比例积分调节规律可有效地消除稳态误差，所以该系统以采用 PI 调节器为好，具体实施时还须对调节器参数作认真修正，以使系统的动态响应时间达到最佳。

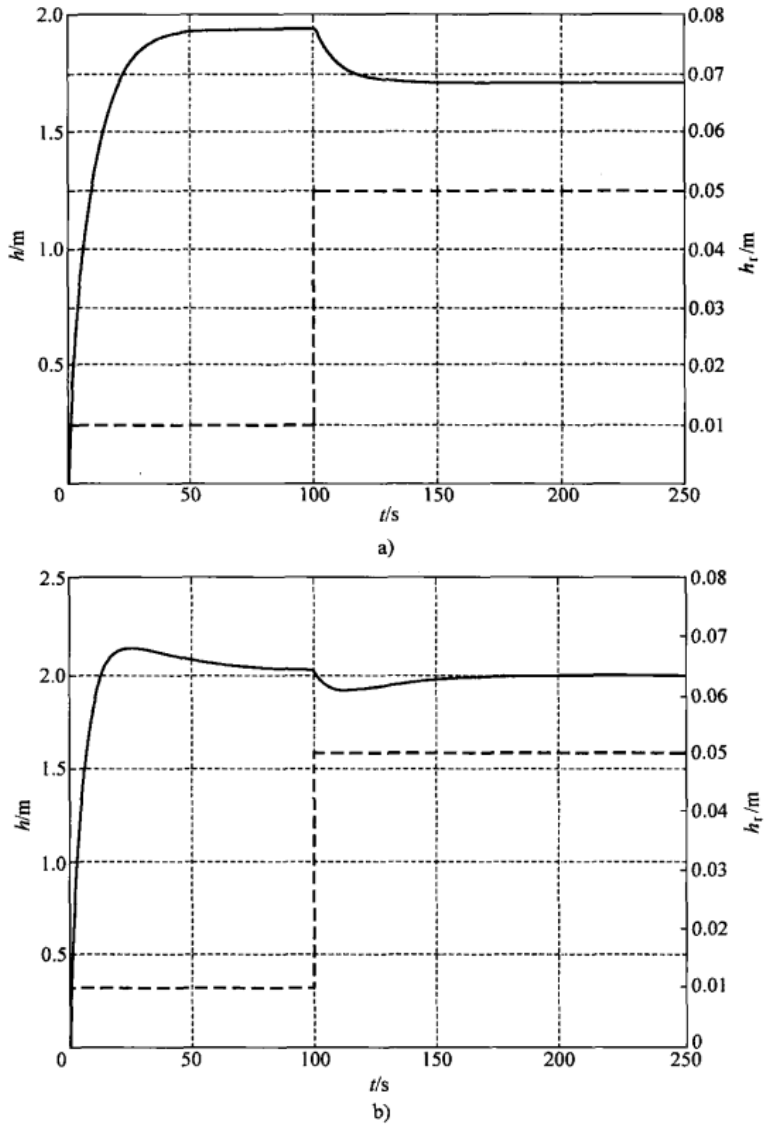


图 5-12 “水箱系统”液位控制仿真结果

a）采用比例调节时的情况 b）采用比例积分调节时的情况

第四节 一阶倒立摆系统的双闭环模糊控制方案

一、引言

在人类自然科学的发展历史上，人们总是以追求事物的精确描述为目的来进行研究，并取得了大量的成果。随着科学技术的进步，在社会生产和生活中存在的大量的不确定性开始引起人们的注意。有关模糊不确定性的研究直到 1965 年，美国的 L. A. Zadeh 教授首次提出模糊集合的概念之后才得到广泛开展。

“模糊”是与“精确”相对而言的概念，模糊性普遍存在于人类的思维和语言交流中，

是一种不确定性的表现。随机性则是客观存在的另一类不确定性，两者虽然都是不确定性，但存在本质上的区别。模糊性主要是人对概念外延的主观理解上的不确定性，而随机性则主要反映客观上的自然的不确定性，即对事件或行为的发生与否的不确定性。

一阶直线倒立摆系统是一个典型的“快速、多变量、非线性、自不稳定系统”，将模糊控制方法应用于一阶倒立摆系统的控制问题，能够发挥模糊控制在非线性系统控制、复杂对象系统控制方面的优势，简化设计，提高控制系统的鲁棒性。

二、模糊理论中的几个基本概念

（一）模糊集合与隶属函数

模糊集合的概念于1965年由Zadeh教授首次提出，它是一个没有精确边界的集合，其定义为：

设X是对象x的集合，x是X的任意元素。X上的模糊子集A定义为一组有序对：$A=\{(x,\mu_A(x)) \mid x\in X\}$，其中，$\mu_A(x)$，被称为模糊子集$A$的隶属函数（Membership Function，简称MF）。称X为论域。

隶属函数（MF）定义了一种将输入空间映射到［0，1］的函数关系，常用的隶属函数有三角形、梯形、高斯型、sigmoid型等多种形式。

（二）模糊逻辑操作

与经典集合运算类似，模糊集合之间也存在交、并、补等运算关系。设A，B是论域U上的模糊集合，A与B的交集$A\cap B$、并集$A\cup B$和A的补集$\overline{A}$也是论域U上的模糊集合。设任意元素$u\in U$，则u对A与B的交集、并集和A的补集的隶属函数分别定义如下：

交运算：$\mu_{A\cap B}(u)=\min\{\mu_A(u),\mu_B(u)\}$

并运算：$\mu_{A\cup B}(u)=\max\{\mu_A(u),\mu_B(u)\}$

补运算：$\mu_{\overline{A}}(u)=1-\mu_A(u)$

（三）模糊规则与模糊推理

模糊规则与模糊推理是模糊推理系统的基础，是模糊集合理论最重要的建模工具。

（1）模糊（if－then）规则　在模糊推理系统中，模糊规则以模糊语言的形势描述人类的经验和知识，规则是否能正确反映专家的经验和知识，是否能正确反映对象的特性，直接决定模糊推理系统的性能。模糊（if－then）规则（也称之为模糊蕴含、模糊条件句）的形式通常为“if前件 then后件”即

if x is A ，then y is B

模糊规则的建立对于构造模糊推理系统是非常关键的，建立模糊规则的一般方法主要有三种：①依据专家、操作人员的经验知识建立模糊规则；②依据过程的模糊模型建立模糊规则；③基于学习的方法，通过设计具有自组织、自学习能力的模糊控制器来自动获取模糊规则。

（2）模糊推理　模糊推理也叫近似推理，是从一组规则和已知事实中得出结论的推理过程。模糊推理的执行结果与模糊蕴含操作的定义、模糊合成规则以及连接词“AND”的操作定义等有关，因而有多种不同的算法。

（四）模糊推理系统

通常，在一个实际的模糊控制系统中，模糊推理系统的功能与模糊控制器的功能是等价

的。模糊推理系统（fuzzy system）的结构如图 5-13 所示，在该模糊系统中，包含所有的应用模糊算法和解决所有相关模糊性的必要成分。它由如下四个基本要素组成。

1）知识库（knowledge base）：它包含模糊集、模糊算子的定义和模糊规则映射。

2）推理机制（inference engine）：它执行所有的输出计算。

3）模糊器（fuzzifier）：它将真实的输入值表示为一个模糊集。

4）反模糊器（defuzzifier）：它将输出模糊集转化为真实的输出值，也称之为解模糊。

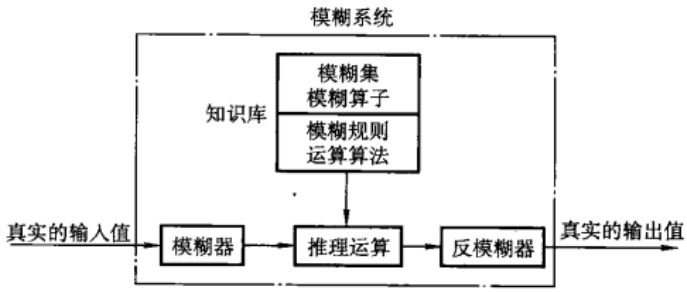


图 5-13 模糊推理系统结构图

与传统的控制方法相比，模糊推理系统有以下特点：①适用于不易获得精确数学模型，或其结构参数不很清楚或难以求得的被控对象，只要求掌握操作人员或领域专家的经验和知识；②它是一种语言变量控制器，控制规则只用语言变量的形式定型表达，构成了被控对象的模糊模型；③系统的鲁棒性强，尤其适用于非线性、时变、滞后系统的控制。

三、一阶倒立摆系统的双闭环模糊控制

（一）问题的提出

一阶倒立摆控制系统共有四个状态变量（θ，$\dot{\theta}$，x，$\dot{x}$），它们均可作为控制器的输入参量。若对这四个输入量定义 5 个模糊子集，则在只有一个模糊控制器的系统中（见图 5-14），由于规则数与输入变量维数成几何关系，模糊规则最多将达到 $5^4=625$ 条，而且每一条规则又由四个前件（条件）和一个后件（结论）组成，这样将使得模糊控制器的设计问题十分复杂，而且在实现过程中对于模糊控制的执行设备（如计算机）的运算速度要求过高，很难实现实时的控制。

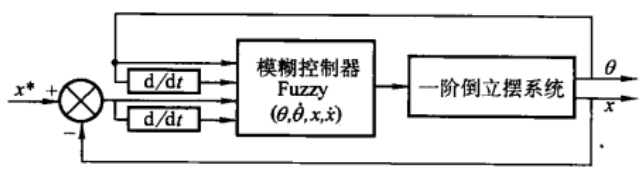


图 5-14 单一模糊控制器系统结构图

为了避免上述的“维数灾难”问题，可以采用如图 5-15 所示的双闭环结构。将系统的

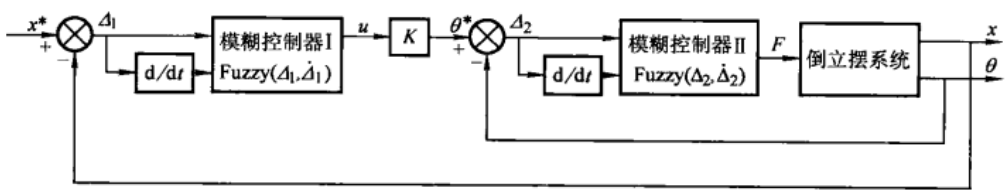


图 5-15 双闭环模糊控制系统结构图

摆角和位置分别作为控制系统内外环控制对象。同样在将每个输入变量定义 5 个模糊子集的情况下，每个控制器的控制规则总数最多只有 $5^2 = 25$ 条，同时，每条规则只有两个前件（条件）和一个后件（结论）。控制规则设计问题将大为简化，控制系统的执行时间也将大大减少，对控制设备的性能要求大为降低。

（二）模糊控制器的设计

（1）隶属函数的定义　由模糊控制理论可知，在进行模糊控制算法的设计之前必须将系统精确量的输入输出转换成对应的语言值，即必须首先确定各个输入输出量的论域及隶属函数。

论域的确定可通过对实物装置的测量（如倒立摆的摆角范围和小车位移范围）、实验辨识或者通过经验知识确定（角速度和线速度范围）。

对于隶属函数形式的选择，为了简化运算、缩短控制周期，对输入、输出变量的隶属函数均采用较为简单的形式。输入变量的隶属函数定义成三角形或梯形隶属函数，输出变量则采用单点隶属函数。经过这样的定义后系统的模糊化和解模糊过程将变得十分简单。各变量隶属函数的具体定义如图 5-16 所示。

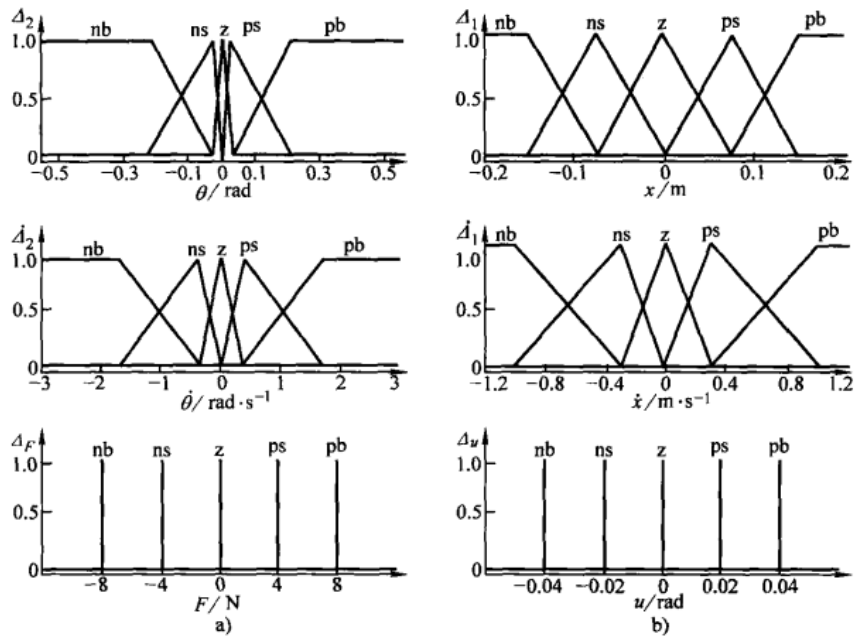


图 5-16　隶属函数定义

a）内环隶属函数　b）外环隶属函数

（2）模糊控制规则　模糊控制规则是模糊控制器的核心，它是将操作者的实践经验加以总结，而得到的一条条模糊条件语句的集合。在一阶倒立摆双闭环模糊控制系统中，内、外环控制器的输入量均为偏差及其对应的偏差变化率，输出为控制量。在这种情况下，可以借助经验公式设计控制规则：将五个模糊子集 nb，ns，z，ps，pb 分别用数值 −2，−1，0，1，2 代换，则结论的数字大约为两个前件数值代数和的一半。依此经验公式选定控制规则的初值，再经实验调整，即可得到系统内外环的模糊控制规则集，按照上述方法得到的一阶倒立摆系统双闭环模糊控制内外环模糊控制器的具体规则见表 5-1、表 5-2。

表 5-1 内环模糊控制规则

Δ_F		Δ_2				
		nb	ns	z	ps	pb
$\dot{\Delta}_2$	nb	nb	nb	nb	ns	z
	ns	ns	nb	ns	z	ps
	z	nb	ns	z	ps	pb
	ps	ns	z	ps	pb	pb
	pb	ze	ps	pb	pb	pb

表 5-2 外环模糊控制规则

Δ_u		Δ_1				
		nb	ns	z	ps	pb
$\dot{\Delta}_1$	nb	nb	nb	nb	ns	z
	ns	ns	nb	ns	z	ps
	z	nb	ns	z	ps	pb
	ps	ns	z	ps	pb	pb
	pb	ze	ps	pb	pb	pb

(3) 解模糊　解模糊过程是模糊化过程的逆过程，即将由模糊控制算法得到的模糊控制输出语言值，依据输出量隶属函数和解模糊规则转换成对应的精确化输出量。

由于在一阶倒立摆双闭环模糊控制系统中，内外环模糊控制器的输出量的隶属函数均为单点集，所以这里采用重心法解模糊的单点公式作为解模糊算法。根据重心法解模糊的单点计算公式可以方便的推导出控制系统内外环解模糊的计算公式分别为

内环解模糊运算公式

$$F = \sum_{i=1}^{25} \mu f_i \Big/ \sum_{i=1}^{25} \mu_i$$

外环解模糊运算公式

$$u = \sum_{i=1}^{25} \omega_i s_i \Big/ \sum_{i=1}^{25} \omega_i$$

四、仿真实验

(一) MATLAB 模糊逻辑工具箱

使用 MATLAB 模糊逻辑工具箱中的图形界面工具（GUI）可以方便地建立起模糊逻辑系统。MATLAB 模糊逻辑工具箱有五个主要的图形界面工具（GUI），可以用来方便快捷地建立、编辑和观察模糊推理系统。这五个 GUI 工具中包括三个编辑器：模糊推理系统（FIS）编辑器、隶属函数编辑器、模糊规则编辑器；两个观察器：模糊规则观察器和输出曲面观察器。而且这五个 GUI 工具之间为动态连接——使用中任何一个 GUI 工具中的参数被修改，其他打开的 GUI 工具中的相应参数或性质也将自动改变。

下面首先介绍这五个 GUI 工具的功能及使用方法，以便读者在具体应用时参考。

(1) FIS 编辑器　FIS 编辑器用来处理系统的高级问题，如确定输入、输出变量的数目及其名称，模糊控制器的命名，选择模糊推理方法等。

要打开 FIS 编辑器，可在 MATLAB 工作区键入命令“fuzzy”，也可利用命令“fuzzy + 系

统名”直接打开一个已有的模糊推理系统。打开的 FIS 编辑器如图 5-17 所示。下面介绍 FIS 编辑器的各部分功能。

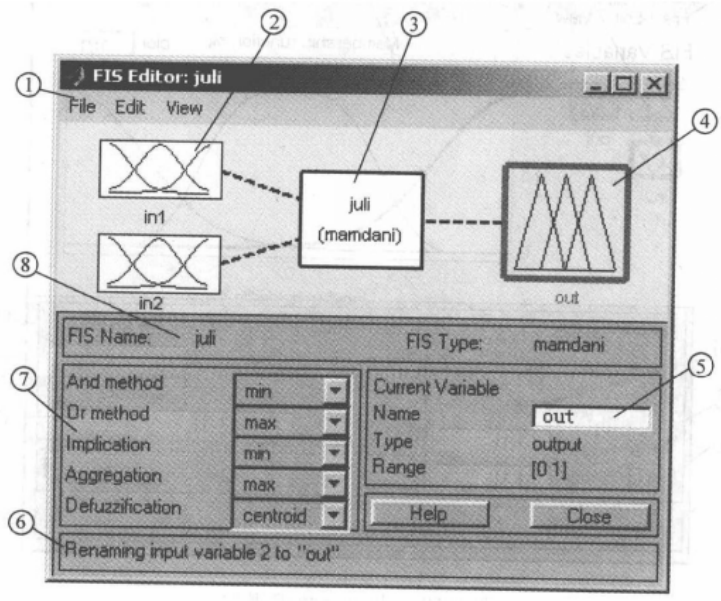


图 5-17　FIS 编辑器图形界面

①　File（文件）菜单，在此可进行打开、保存、编辑模糊系统等操作。

②　双击输入变量图标，可打开隶属函数编辑器，定义输入变量的隶属函数。

③　双击系统图标，可打开模糊规则编辑器。

④　双击输出变量图标，可打开隶属函数编辑器，定义输出变量的隶属函数。

⑤　文本框，可对输入输出变量进行命名或改名。

⑥　状态栏，显示上一步进行的操作。

⑦　下拉菜单，用于选择模糊推理方法。

⑧　显示系统名称，要更改系统名称，可在 File（文件）菜单下选择“Save as …”进行。

（2）隶属函数编辑器　先以下面三种方式中的任何一种打开隶属函数编辑器。

1）拉下 View 菜单项，选定“Edit Membership Functions…”。

2）双击输入/输出变量图标。

3）在命令行键入“mfedit”。

打开的隶属函数编辑器如图 5-18 所示。其各区域功能如下：

①　File（文件）菜单，在此可进行打开、保存、编辑模糊系统等操作。

②　变量区，显示所有已定义的输入输出变量。单击某变量，使其成为当前变量，就可编辑其隶属函数。

③　绘图区，显示当前变量的隶属函数。

④　单击选中一条隶属函数曲线，就可以编辑修改隶属函数的名称、类型、属性及参数。

⑤　文本框，可改变当前隶属函数的名称。

⑥　下拉菜单，可用来改变当前隶属函数的类型。

⑦　文本框，可以改变当前隶属函数的数字参数。

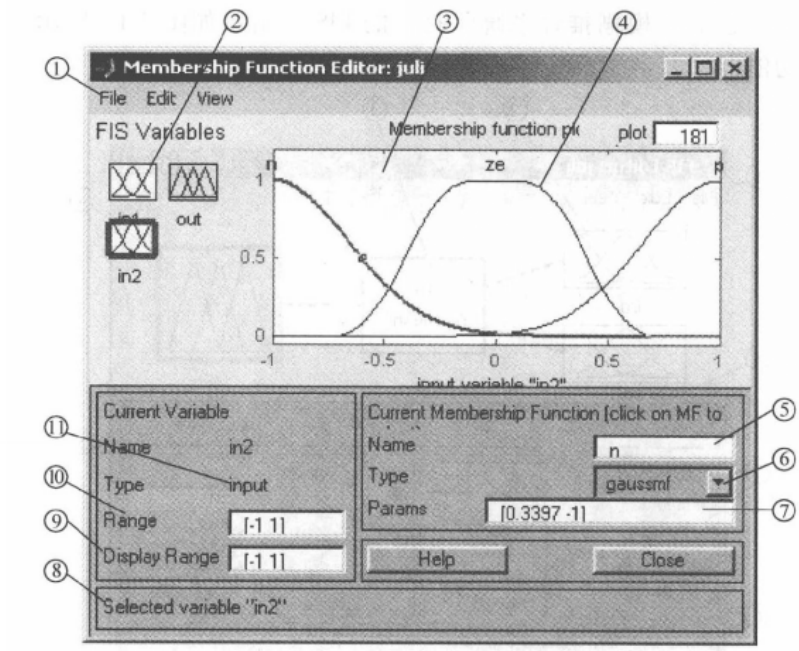


图 5-18 隶属函数编辑器

⑧ 状态栏，显示上一步操作。

⑨ 文本框，可设置当前图形的显示范围。

⑩ 文本框，设置当前变量范围。

⑪ 显示当前变量的名称和类型。

（3）模糊规则编辑器 有两种方法可以调用模糊规则编辑器，一是在 FIS 编辑器中的 View 菜单中选定“Edit rules...”命令，二是在命令行键入“ruleedit”。

图 5-19 中的模糊规则编辑器各部分功能如下：

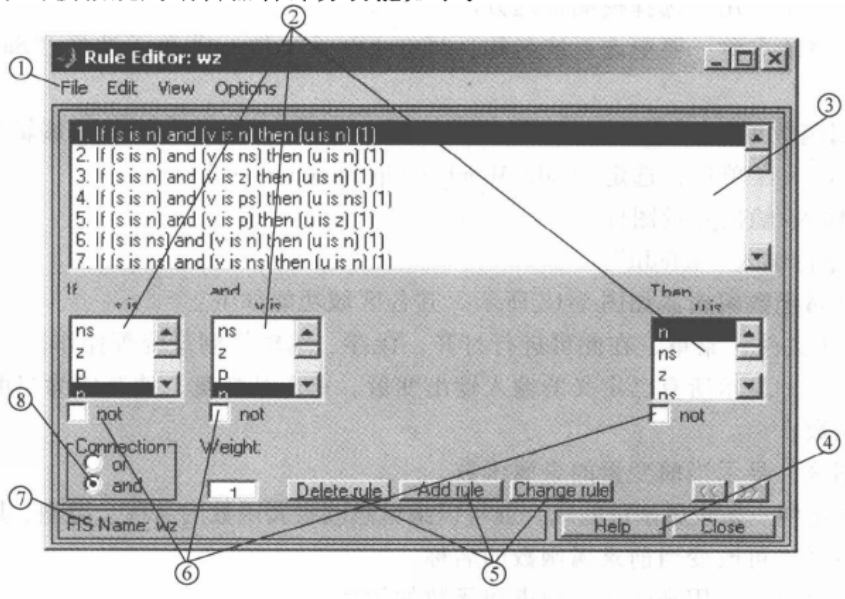


图 5-19 模糊规则编辑器

① File（文件）菜单，在此可进行打开、保存、编辑模糊系统等操作。

② 输入输出选择框。

③ 模糊规则显示区。

④ 这两个按钮可调用编辑器的使用帮助和关闭窗口。

⑤ 这三个按钮用于删除、修改和增加模糊规则。

⑥ 选中此按钮模糊规则的输入输出表述将为“非”。

⑦ 状态栏，显示上一步操作。

⑧ 选择模糊规则中个输入间的“与”、“或”关系。

此外，在工作区中键入“ruleview”或从 FIS 编辑器的 View 菜单中选择“View rules...”可进入模糊规则观察器，观察模糊推理图即模糊系统的推理过程是否与预期的相同。在工作区中键入“surfview”或从 FIS 编辑器的 View 菜单中选择“View surface...”可进入输出曲面观察器，从中可以观察输入区间与输出区间的整体对应关系。

（二）一阶倒立摆系统数字仿真模型的建立

下面讨论如何利用 MATLAB 的 GUI 工具及 Simulink 的模糊控制工具箱（Fuzzy Logic Toolbox）进行一阶倒立摆模糊控制系统仿真试验的问题。

（1）建立 Simulink 仿真模型　利用 Simulink 基本模块库和 Fuzzy Logic Toolbox 建立如图 5-20 所示的仿真模型。其中，包含倒立摆模型（inverted pendulum）、内外环模糊控制器（Fuzzy Logic Controller）、差分模块（difference）、增益调整（K）、数据文本输出模块（仿真数据输出到文本 mohu. mat）。

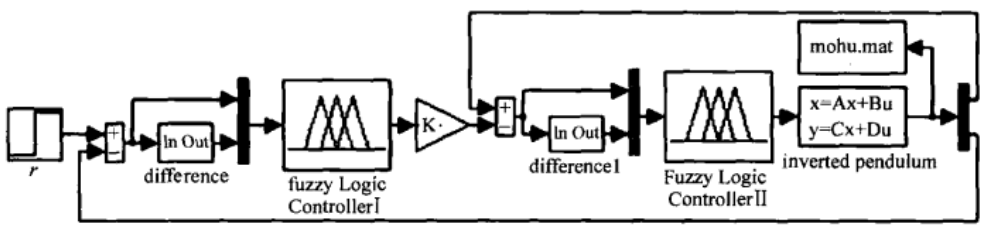


图 5-20　系统仿真模型

作为受控对象的一阶倒立摆的模型（inverted pendulum），这里采用的是系统数学模型的状态空间表达式。模型参数依据实物系统选定为：滑块质量 $M=0.6\text{kg}$；倒摆振子质量 $m=0.085\text{kg}$；倒摆长度 $2L=0.42\text{m}$；重力加速度取 $g=9.8\text{m/s}^2$，摩擦因数 $D=0.01\text{N/m}\cdot\text{s}^{-1}$。

系统内环模糊控制器（Fuzzy Logic Controller Ⅱ）和外环模糊控制器（Fuzzy Logic Controller Ⅰ）中的模糊控制规则定义见表 5-1 和表 5-2。各输入输出量的隶属函数定义如图 5-16 所示；内外环模糊控制器均采用 Mamdani 型模糊推理算法及“极大-极小”合成规则进行模糊推理。

系统仿真模型中的差分模块子系统（difference）的内部结构如图 5-21 所示，差分采样时间为 10ms。采用差分计算代替微分模块是出于以下考虑：

1）考虑仿真时间问题，采用微分计算系统仿真过程的计算量将变得很大，特别是在系

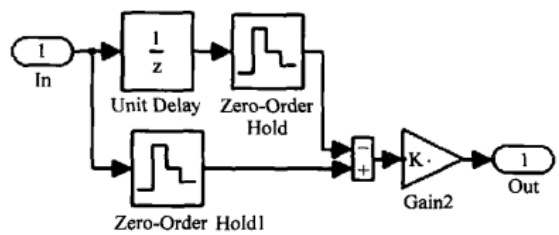


图 5-21　差分模块子系统的内部结构

统的暂态过程中，有可能出现某一瞬时系统的微分量很大导致微分计算困难。

2）考虑与实物实验的一致性。在实物平台上由于采用离散的控制器（计算机、DSP等），实际系统的变量变化率求取同样是采用差分的办法。

差分模块子系统（difference）中采用的零阶采样保持器（Zero - Order Hold）起到了将离散输出量“连续化”的作用。

（2）建立模糊逻辑系统　利用前述的模糊控制理论知识，分别建立系统内外环模糊逻辑控制系统，并将它们分别命名，如“nh”和“wh”，再分别双击 Simulink 系统仿真模型中的内外环模糊控制器模块，在弹出对话框中分别键入两个系统的名称。这样就建立 MATLAB 模糊逻辑系统与 Simulink 系统仿真模型之间的联系，继而可运行 Simulink 仿真。

（3）利用仿真数据文本绘制仿真曲线　运行 Simulink 仿真结束后，仿真结果被保存到仿真数据文本 mohu. mat 中，可在 MATLAB 中建立一个 m 文件调用此数据文件绘制仿真结果曲线。

以下为 m 文件的一个实例。

```
% Inverted pendulum
% signals recuperation
load mohu. mat
t = signals (1,:);
q = signals (2,:);
x = signals (3,:);
% drawing x (t) and theta (t) response signals
figure (1)
hf = line (t, q (:));
grid on
xlabel ('时间 (s)')
ylabel ('摆角 (rad)')
axis ( [0 25 -0.02 0.14])
axet = axes ('Position', get (gca,'Position'), …
             'XAxisLocation','bottom', …
             'YAxisLocation','right','Color','None', …
             'XColor','k','YColor','k');
ht = line (t, x,'color','r','parent', axet);
ylabel ('位置 (m)')
axis ( [0 25 -0.02 0.14])
title ('\ theta (t) and x (t) Response to a step input')
```

（三）仿真实验结果

利用仿真数据文本 mohu. mat 中的数据，可以绘制相关的仿真结果曲线。当系统的初始状态为系统原点，位置给定位为 0.1m，仿真实验结果及摆角响应曲线放大图如图 5-22 所示。由图中可见，闭环系统运行稳定，并且具有较高的稳态精度。过渡过程中摆角的摆动幅度 $< \pm 0.03$rad，说明系统运行平稳。

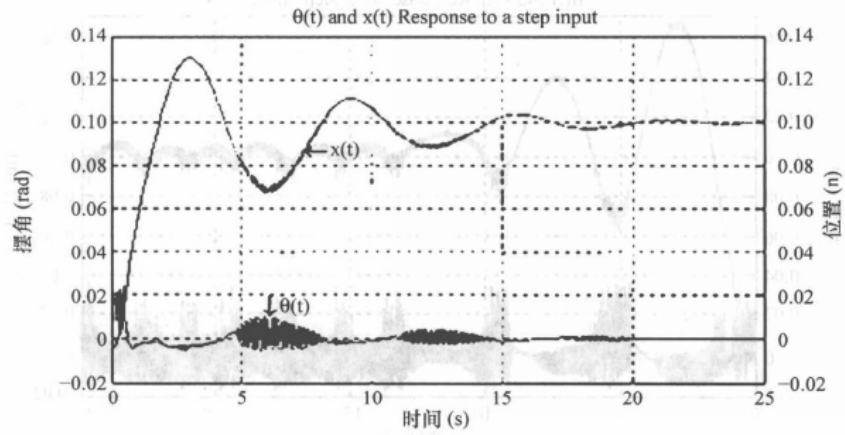


图 5-22　仿真实验结果及摆角响应曲线放大图

a）仿真实验结果　b）摆角响应曲线放大图

实际上，控制系统的参数、规则等（如输入输出隶属函数的形式和参数、模糊规则的定义）选择都不是唯一的，可以通过反复试验和经验知识来寻求满足系统性能实际要求的具体控制参数。当系统的控制参数改变时，系统的特性也随之改变。比如，将以上实验中内环输出量的隶属函数改变为图 5-23 的形式，则仿真结果如图 5-24 所示。

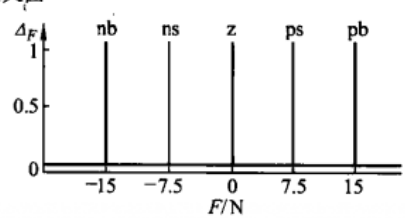


图 5-23　内环输出量隶属函数

可见，系统仍为稳定，但系统的超调量明显变大，稳态性能指标与前面的结果也有较大差距，所以，不同隶属函数的选取对控制系统的性能影响较大。

为了进一步验证控制系统的鲁棒性，适当改变被控对象（一阶倒立摆系统）的模型参数，在模糊控制器参数设置不变的情况下进行仿真实验。图 5-25、图 5-26 和图 5-27 分别为对象模型参数作不同变化时的仿真结果。它们对应的模型参数设置见表 5-3。

表 5-3　模型参数设置

情况	1	2	3
小车质量 M/kg	0.6	0.6	0.6
摆杆质量 m/kg	0.17	0.085	0.25
摆杆长度 $2L$/m	0.42	0.84	1.2

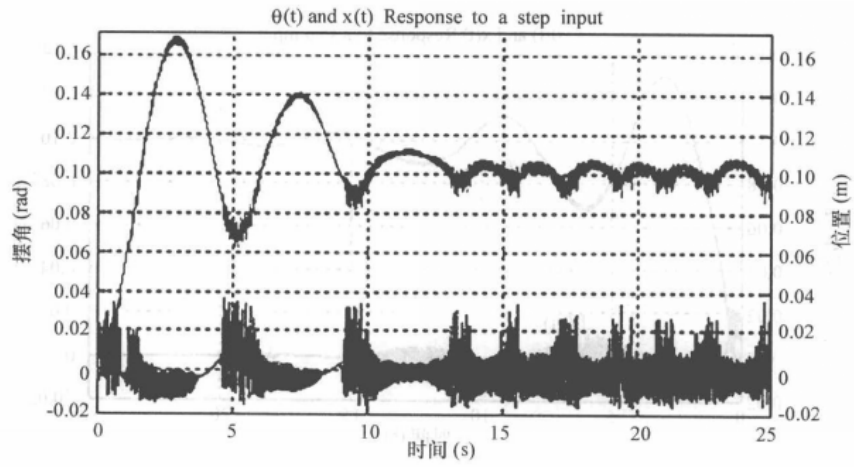


图 5-24 仿真实验结果 2

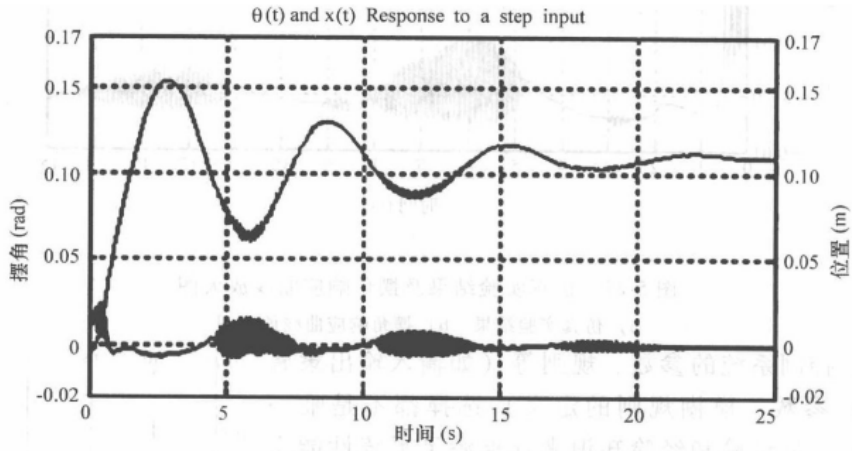


图 5-25 仿真实验结果 1

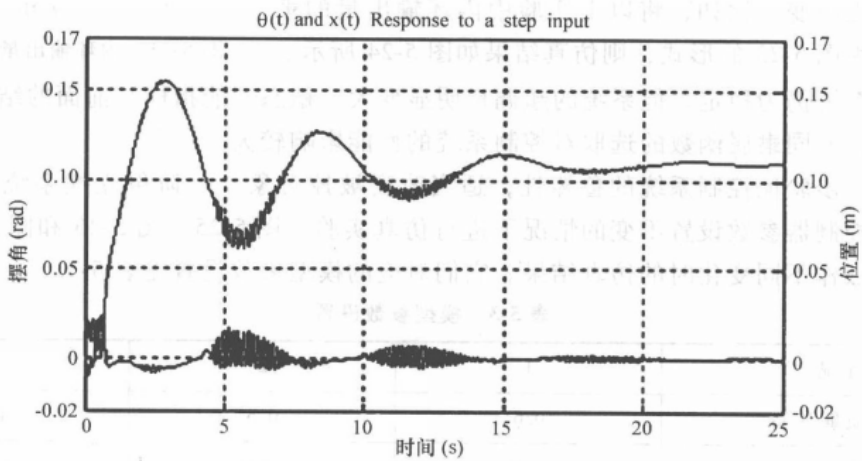


图 5-26 仿真实验结果 2

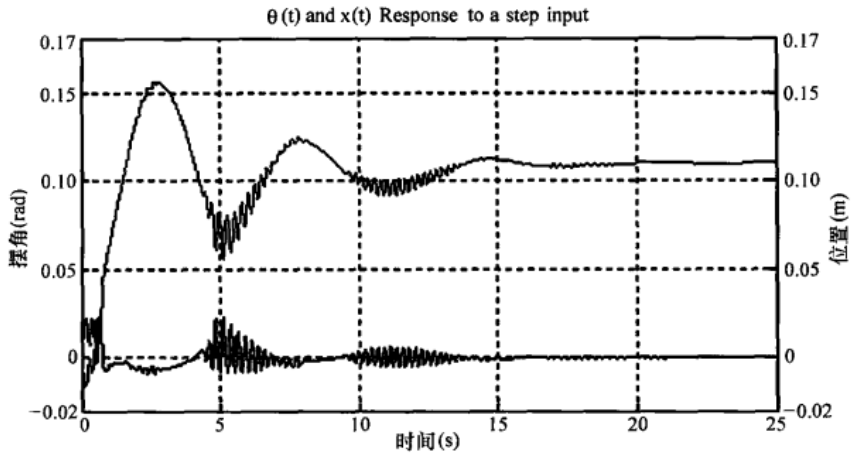


图 5-27　仿真实验结果 3

由图 5-25、图 5-26 和图 5-27 仿真结果曲线可见，系统的模糊控制器具有很强的鲁棒性。当控制对象（一阶倒立摆）的参数变化时，随着参数变化的增大，系统响应的超调量和稳态误差稍有加大，而控制器仍然能够有效控制系统的稳定。当摆杆质量和长度均达到实际系统的 2.5 倍时（见图 5-27），模糊控制器依然能够控制系统的平衡，且系统的稳态误差 <0.01m。模糊控制器鲁棒性强这一特性对于实现一阶倒立摆实物系统的稳定控制是十分有利的。

五、结论

1）双闭环模糊控制方案能够有效控制一阶倒立摆系统的稳定，并且实现小车的有效定位，而且双闭环结构使得系统的控制规则和执行时间大幅减少，大大降低了系统设计和实现难度。

2）仿真实验证明，一阶倒立摆系统的双闭环模糊控制方案具有较强的鲁棒性。

3）本节内容对于一类有多个反馈量、非线性、自不稳定的系统的控制问题具有一定的参考价值。

第五节　基于单位功率因数的 PWM 整流器控制系统设计

一、引言

整流器作为电力电子设备的前端电路，应用极其广泛。传统的整流装置采用二极管不控整流或晶闸管相控整流方式，具有网侧电流谐波大、功率因数低等缺点，为电网带来了严重的谐波和无功功率污染问题。

三相电压型 PWM 整流器是一种新颖的采用全控型器件的整流器，借助于控制手段，它能够实现单位功率因数运行，具有网侧电流畸变小、输出电压可调、效率高及能量可双向流动等优点，越来越受到人们的广泛关注。

PWM 整流器优良性能的获得依赖于控制系统的设计，由于 PWM 整流器是非线性系统，采用线性系统理论设计的控制算法存在着控制器参数整定困难、过于依赖系统参数、大范围

扰动不稳定等问题。因此，深入研究三相电压型 PWM 整流器的非线性控制方法，对于提高整流器控制性能、解决电网谐波和无功污染问题、提高电能质量及电能利用效率具有重要意义。在本节中，我们将采用滑模变结构控制方案，实现对三相电压型 PWM 整流器的单位功率因数的控制。

二、滑模变结构控制

变结构控制理论诞生于 20 世纪 50 年代末，作为一种非线性控制理论，与其他控制方法相比，具有控制规律简单，对系统的数学模型精确性要求不高，可以有效地平衡动、静态之间的矛盾以及具有强鲁棒性等优点，近年来已被广泛应用于处理一些复杂的非线性、时变、多变量耦合及不确定系统，如伺服电动机驱动、机器手控制以及飞船控制系统的控制。下面先对滑模变结构控制的一些基本概念进行阐述。

（一）滑动模态

滑模变结构控制是变结构控制系统的一种控制策略。这种控制策略与常规控制的根本区别在于控制的不连续性，即一种使系统结构随时变化的开关特性。该控制特性可以迫使系统在一定条件下沿规定的状态轨迹作小幅度、高频率上下运动，即滑动模态或“滑模”运动。这种滑动模态是可以设计的，且与系统的参数及扰动无关。这样，处于滑模运动的系统自然就具有很好的鲁棒性。

（二）切换函数及切换面

在滑模变结构控制中，需要通过开关的切换，改变系统在状态空间中的切换面 $s(x)=0$ 两边的结构。开关切换的法则称为控制策略，它保证系统具有滑动模态。此时，分别把 $s=s(x)$ 及 $s(x)=0$ 叫作切换函数及切换面。

（三）滑模变结构控制的数学描述

滑模变结构控制可表述成如下形式：

设有一非线性控制系统

$$\dot{x}=f(x,u,t) \qquad x\in R^n, u\in R^m, t\in R$$

需要确定切换函数向量

$$s(x), s\in R^n$$

具有的维数一般情况下等于控制的维数，并且寻求变结构控制

$$u_i(x)=\begin{cases} u_i^+(x) & \text{当 } s_i(x)>0 \\ u_i^-(x) & \text{当 } s_i(x)<0 \end{cases}$$

这里变结构体现在 $u^+(x)\neq u^-(x)$，使其满足以下条件：

1）满足到达条件：切换面 $s_i(x)=0$ 以外的相轨迹将于有限时间内到达切换面。

2）切换面是滑动模态区，且滑动运动渐进稳定，动态品质良好。

满足以上条件的控制叫作滑模变结构控制，其基本原理如图 5-28 所示。

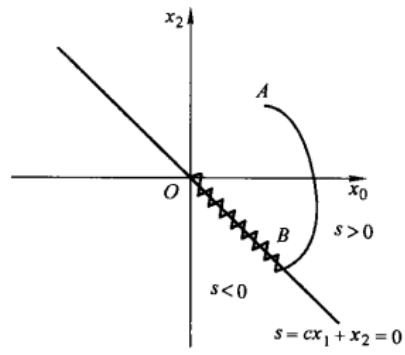


图 5-28 滑模变结构控制的基本原理

（四）滑模变结构控制系统的设计方法

滑模变结构控制系统的设计步骤可概括如下：

1）选择滑模面参数，构成希望的滑动模态。

2）求取不连续控制 u^*，保证在切换平面 $s=0$ 上的每一点存在滑动模态，这一平面就被认为是滑动面。

3）控制必须让状态进入滑动面。

滑动模态三要素（即存在、稳定、进入）是靠切换面与控制两者来保证的，一旦切换面选定，则问题转入控制的求取。求取标量变结构控制，要从滑动模态的存在条件出发，即从 $s\dot{s}<0$ 这个关系式出发。按这个关系式所求得的控制，往往是不等式。在选取时，可充分考虑滑动模态的进入条件。对于滑动模态的稳定性问题，由于在选择切换面时已经考虑过，因此不需要再进行讨论[97,98]。

三、系统建模与模型验证

在第二章第二节“三相电压型 PWM 整流器系统控制问题”中，我们给出了三相静止坐标系下三相电压型 PWM 整流器的数学模型（为表述方便，在本小节中将“三相电压型 PWM 整流器”统一简称为“PWM 整流器”），这里我们用“必要条件法”对所建模型进行验证。所谓必要条件法，就是所进行的模型验证实验的结果是依据经验可以判定的，其正确性的结果是正确的模型所应具备的必要性质。

在 MATLAB/Simulink 环境下，按照式（2-52）所示的 PWM 整流器数学模型搭建其仿真模型，如图 5-29 所示。这里函数 Fun、Fun1、Fun2、Fun3 和 Fun4 的表达式分别为

```
Fun:(u[4]-u[11]*u[1]-u[10]*u[7]+u[10]*(u[7]+u[8]+u[9])/3)/u[12]
Fun1:(u[5]-u[11]*u[2]-u[10]*u[8]+u[10]*(u[7]+u[8]+u[9])/3)/u[12]
Fun2:(u[6]-u[11]*u[3]-u[10]*u[9]+u[10]*(u[7]+u[8]+u[9])/3)/u[12]
Fun3:(u[1]*u[7]+u[2]*u[8]+u[3]*u[9]-u[10]/u[14])/u[13]
```

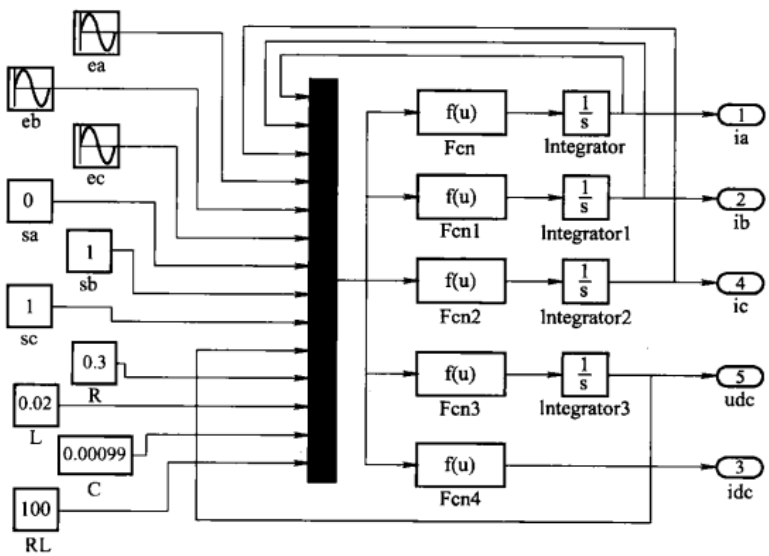


图 5-29　模型验证仿真模型

Fcn4:u[1] * u[7] + u[2] * u[8] + u[3] * u[9]

系统模型方程中的常量 R，L，C，R_L 是可变参数，其可在模型外部进行灵活设置，这里取 $R=0.3\Omega$，$L=20\text{mH}$，$C=990\mu\text{F}$，$R_L=200\Omega$。由 $i_{dc}=i_a s_a+i_b s_b+i_c s_c$，可得不同开关模式时的 i_{dc} 值见表 5-4。

表 5-4 PWM 整流器不同开关模式时 i_{dc} 的取值

$s_c s_b s_a$	001	010	011	100	101	110	111	000
i_{dc}	i_a	i_b	$-i_c$	i_c	$-i_b$	$-i_a$	0	0

为了检验 PWM 整流器数学模型是否与实际系统相符，下面设计了三个仿真实验，见表 5-5。

表 5-5 模型验证仿真实验

序号	实验条件	判定依据
实验一	$s_a=s_b=s_c=0$	$i_{dc}=0$
实验二	$s_a=1$，$s_b=s_c=0$	$i_{dc}=i_a$
实验三	$s_a=0$，$s_b=s_c=1$	$i_{dc}=-i_a$

模型验证的仿真实验结果如图 5-30 所示。由实验结果可以看出，该模型行为与理论分析完全符合，因而可在一定程度上证明所建模型的正确性。

四、基于滑模变结构控制的 PWM 整流器控制系统设计

在第二章第二节的“三相电压型 PWM 整流器系统控制问题”中，给出了 dq 同步旋转坐标系下三相电压型 PWM 整流器的数学模型，即

$$\begin{cases} L\dfrac{\mathrm{d}i_d}{\mathrm{d}t}=-Ri_d+\omega Li_q-s_d u_o+e_d \\ L\dfrac{\mathrm{d}i_q}{\mathrm{d}t}=-\omega Li_d-Ri_q-s_q u_o+e_q \\ C\dfrac{\mathrm{d}u_o}{\mathrm{d}t}=s_d i_d+s_q i_q-\dfrac{u_o}{R_L} \end{cases} \tag{5-16}$$

这里，所设计的 PWM 整流器滑模变结构控制系统将采用双闭环结构，内环为电流环，控制目标是实现网侧电流正弦化，保证 PWM 整流器高功率因数运行；外环为电压环，控制目标是实现输出直流电压快速调节，且无稳态误差。据此，控制系统设计包括电流内环控制器设计和电压外环控制器设计两部分，控制系统结构如第二章第二节“三相电压型 PWM 整流器系统控制问题”中的图 2-21 所示。

（一）电流内环控制器的设计

在电流内环控制器设计时，暂不考虑式（5-16）中的电压方程，即此时被控对象的数学方程为

$$\begin{cases} \dfrac{\mathrm{d}i_d}{\mathrm{d}t}=\dfrac{1}{L}e_d-\dfrac{1}{L}s_d u_o+\omega i_q-\dfrac{R}{L}i_d \\ \dfrac{\mathrm{d}i_q}{\mathrm{d}t}=\dfrac{1}{L}e_q-\dfrac{1}{L}s_q u_o-\omega i_d-\dfrac{R}{L}i_q \end{cases} \tag{5-17}$$

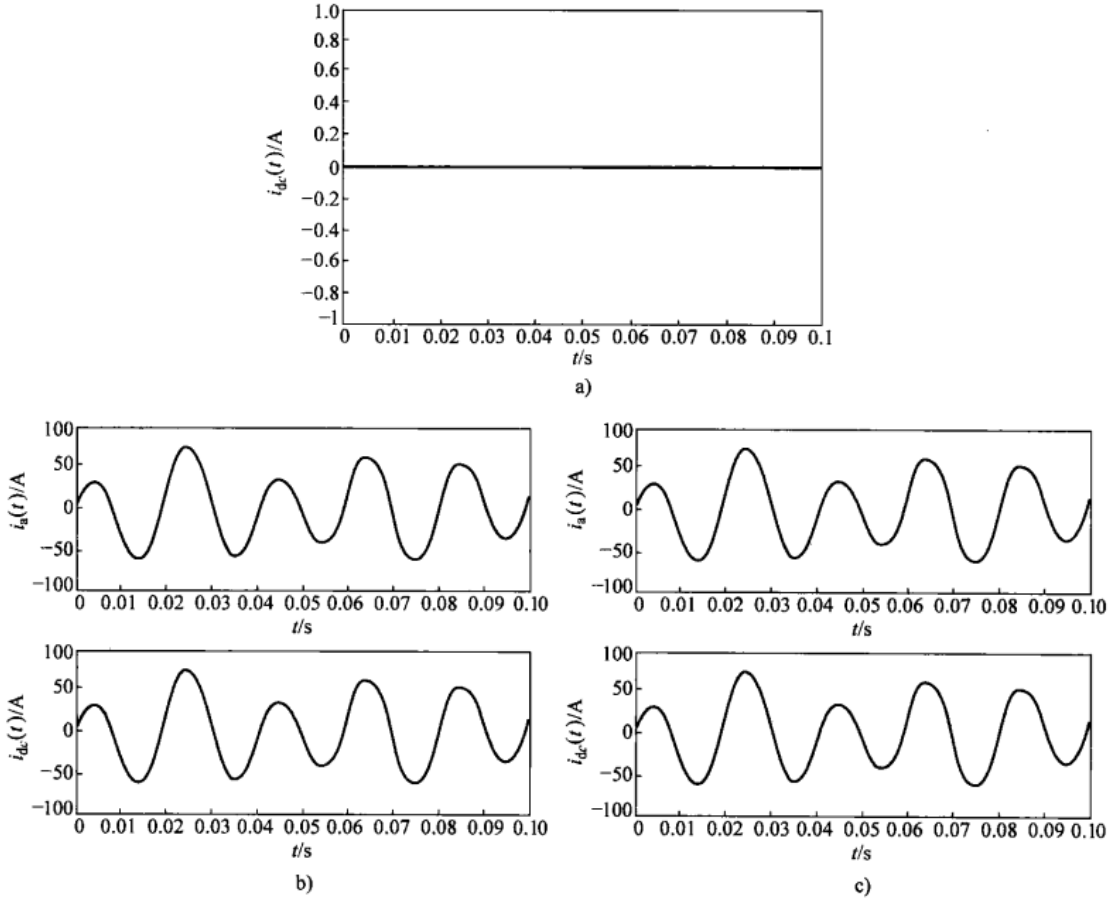


图 5-30　模型验证的仿真实验结果

a）实验一的仿真实验结果　b）实验二的仿真实验结果　c）实验三的仿真实验结果

在此数学模型基础上，我们将设计一个多输入向量滑模变结构控制器，使 i_d 和 i_q 都能快速达到给定值。

1）定义两个滑模面：

$$\begin{cases} s_1 = \alpha_1(i_d - i_d^*) \\ s_2 = \alpha_2(i_q - i_q^*) \end{cases} \tag{5-18}$$

式中，α_1，α_2 均是正实数；i_d^*，i_q^* 分别是 i_d，i_q 的给定值。

2）采用等速趋近率：

$$\frac{ds}{dt} = -\varepsilon \operatorname{sgn}(s) \tag{5-19}$$

式中，ε 为正实数，则可保证 $s \cdot \dot{s} < 0$，满足广义滑模条件：

$$\begin{cases} \dot{s}_1 = -W_1 \operatorname{sgn}(s_1) \\ \dot{s}_2 = -W_2 \operatorname{sgn}(s_2) \end{cases} \quad W_1 > 0, W_2 > 0 \tag{5-20}$$

3）求出控制率。

因为

$$\begin{cases} \dot{s}_1 = \alpha_1 \dot{i}_d = \alpha_1 \left(\dfrac{1}{L} e_d - \dfrac{1}{L} u_o s_d + \omega i_q - \dfrac{R}{L} i_d \right) \\ \dot{s}_2 = \alpha_2 \dot{i}_q = \alpha_2 \left(\dfrac{1}{L} e_q - \dfrac{1}{L} u_o s_q - \omega i_d - \dfrac{R}{L} i_q \right) \end{cases} \tag{5-21}$$

所以由式（5-20），可得电流内环控制器的控制率如下：

$$\begin{cases} s_d = \dfrac{L}{u_o} \left[\dfrac{1}{L} e_d - \dfrac{R}{L} i_d + \omega i_q + \dfrac{W_1}{\alpha_1} \mathrm{sgn}(s_1) \right] \\ s_q = \dfrac{L}{u_o} \left[\dfrac{1}{L} e_q - \dfrac{R}{L} i_q - \omega i_d + \dfrac{W_2}{\alpha_2} \mathrm{sgn}(s_2) \right] \end{cases} \tag{5-22}$$

（二）电流内环控制器的改进

由式（5-22）可以看出，电流内环控制器的控制率存在不连续符号函数 sgn(s)，它的存在将带来滑模变结构控制所特有的“抖振”问题。

（1）抖振的产生　在实际工程应用中，滑模变结构控制的开关特性 $u(x) = u^*(x)\mathrm{sgn}[s(x)]$ 是不可能实现的，时间延迟和空间滞后等因素将使得滑动模态表现为高频抖动形式。这种现象称之为抖振。

抖振问题是变结构控制广泛应用的突出障碍，是影响变结构技术发展的重要原因。这是因为有的系统元件不能承受高频切换；有的系统性能上不允许存在抖振；抖振的存在还可能激发系统未建模部分的强迫振荡。于是在实践中，人们尝试采用各种具有准滑动模态的控制系统。所谓“准滑动模态”（或近似滑动模态、伪滑动模态），是指系统的运动轨迹被限制在理想滑动模态的某一 Δ 邻域内的模态。

（2）抖振的抑制　抖振发生的本质原因是由于开关的切换动作造成控制的不连续性。因此，对于一个现实的滑模变结构控制系统，抖振必定存在。我们可以努力去削弱抖振的幅度，使它减少到工程允许的范围内，而无法完全消除它，消除了抖振，也就消除了滑模变结构控制系统的抗干扰能力，由于抖振问题是滑模变结构控制的突出障碍，因此许多学者提出了消抖措施，其中具有代表性的是柔化 sgn(s) 函数法、边界层法和趋近率法等[88]。

这里，我们采用“边界层法”，即用连续的饱和函数 sat(s) 来代替不连续函数 sgn(s)。这种降低抖振的方法，也被称为“具有饱和函数的准滑模伪变结构控制”。由于将继电函数连续化，所以系统不再存在结构变化，故称之为“伪变结构”。饱和函数 sat(s) 可写为如下形式：

$$\mathrm{sat}(s) = \begin{cases} +1 & s > \Delta \\ ks & |s| \leqslant \Delta \\ -1 & s < -\Delta \end{cases} \tag{5-23}$$

式中，Δ 被称为“边界层”，且 $\Delta \cdot k = 1$。sgn(s) 和 sat(s) 可用图 5-31 所示曲线表示。

因此改进后的电流内环控制器如下所示：

$$\begin{cases} s_d = \dfrac{L}{u_o} \left[\dfrac{1}{L} e_d - \dfrac{R}{L} i_d + \omega i_q + \dfrac{W_1}{\alpha_1} \mathrm{sat}(s_1) \right] \\ s_q = \dfrac{L}{u_o} \left[\dfrac{1}{L} e_q - \dfrac{R}{L} i_q - \omega i_d + \dfrac{W_2}{\alpha_2} \mathrm{sat}(s_2) \right] \end{cases} \tag{5-24}$$

（三）电压外环控制器的设计

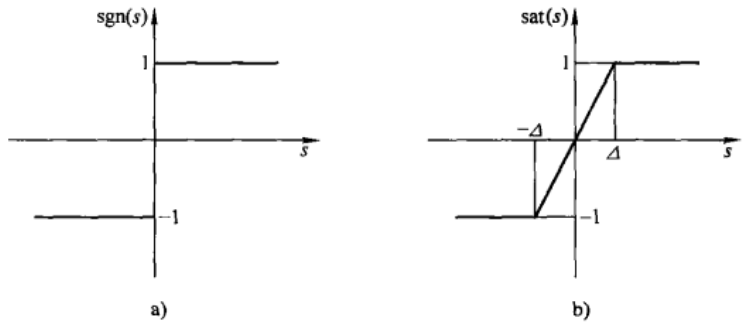


图 5-31　sign 函数及其饱和函数连续化曲线

a）sign 函数曲线　b）饱和函数曲线

PWM 整流器控制系统的电压外环通常采用 PI 控制算法，使得输出直流电压无稳态误差。由于控制系统采用双闭环结构，因此根据电流内环的闭环传递函数，即可设计电压外环 PI 控制器。可将电流内环和 PWM 调制环节等效为一个一阶惯性环节[64]：

$$W_{ci}(s) = \frac{0.75}{3T_s s + 1} \tag{5-25}$$

其中，T_s 为电流内环采样周期，也就是 PWM 开关周期。电压外环控制系统结构如图 5-32 所示。

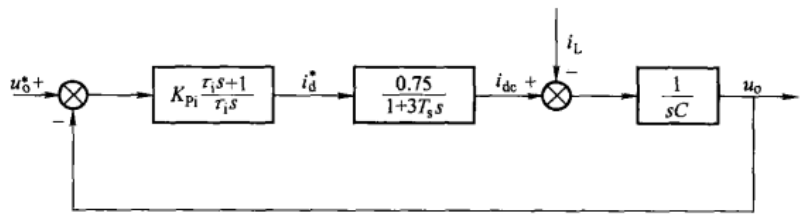


图 5-32　电压外环控制系统结构

这里，电压外环控制器为

$$W_{pi}(s) = K_{pi}\frac{\tau_i s + 1}{\tau_i s} \tag{5-26}$$

则系统开环传递函数为

$$G(s) = \frac{0.75K_{pi}(\tau_i s + 1)}{C\tau_i s^2(3T_s s + 1)} \tag{5-27}$$

由此，得电压环中频宽 h 为

$$h = \frac{\tau_i}{3T_s} \tag{5-28}$$

按照“典型Ⅱ型系统”控制器参数整定方法[6]，可得

$$\frac{0.75K_{pi}}{C\tau_i} = \frac{h + 1}{2h^2(3T_s)^2} \tag{5-29}$$

选择 $h = 5$，可以得到电压外环的 PI 调节器系统参数为

$$\tau_i = 5 \times 3T_s \tag{5-30}$$

$$K_{pi} = \frac{4C}{3T_s} \tag{5-31}$$

在实际工程应用中，需根据上述工程化设计方法，确定电压外环 PI 调节器的基本参数。在此基础上通过适当的参数调整，就可以找到一组系统稳态/动态性能指标兼顾的 PI 参数，这就减少了在实际工程中 PI 参数选择的盲目性。

综上，可以得到如图 5-33 所示的 PWM 整流器滑模变结构控制系统结构。下面对其进行仿真实验。

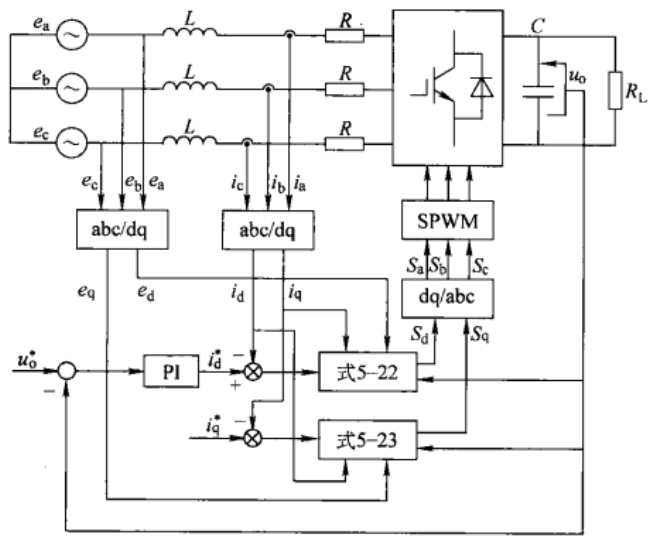


图 5-33 PWM 整流器滑模变结构控制系统结构图

五、仿真实验

（一）电力系统仿真工具箱（SimPowerSystems）

电力系统仿真工具箱是 Matlab/Simulink 环境下专用于 *RLC* 电路、电力电子电路、电动机及其传动系统和电力系统仿真的模型库。该仿真工具箱包括电源模块库（Electrical Sources）、电气元件模块库（Elements）、电力电子元件模块库（Power Electronics）、电动机模块库（Machines）、测量仪器模块库（Measurements）、相量元件模块库（Phasor Elements）和附加电气模块库（Extras）这几大部分，如图 5-34 所示。利用上述模型，可以模拟由电阻、电感和电容组成的 *RLC* 电路，含电力电子元件的整流、逆变和变频等电力电子装置，以及电力系统运行和故障的各种状态。



图 5-34 SimPowerSystems 仿真工具箱的各子模块库

SimPowerSystems 仿真工具箱与其他 Simulink 模块的使用有所不同，SimPowerSystems 仿真工具箱的模块必须连接在回路中使用，因此每个模块都有输入端和输出端，在回路中流动的是电流，并且电流通过每个电气元器件时都产生电压降。其他 Simulink 模块组成的是信号流程，流入流出模块的信号没有特定的物理含义，其含义视仿真模型对象而定。

由 SimPowerSystems 仿真工具箱模块组成的电路和系统可以与 Simulink 模型库中的控制单元连接、组合成开环/闭环控制系统，研究和观察在不同控制方案下系统的稳态和动态响应，为系统分析与设计提供依据。

根据 SimPowerSystems 仿真工具箱，可在 Simulink 环境下建立 PWM 整流器的仿真模型如图 5-35 所示。这一模型将作为被控对象，在控制系统仿真时被使用。

（二）PWM 整流器滑模变结构控制系统的仿真

根据前面设计的控制器表达式，在 MATLAB/Simulink 环境下，建立 PWM 整流器滑模变结构控制系统仿真模型，如图 5-36 所示。

在该仿真模型中，“PI controller”为电压外环 PI 控制器，为保证系统稳定运行，对其输

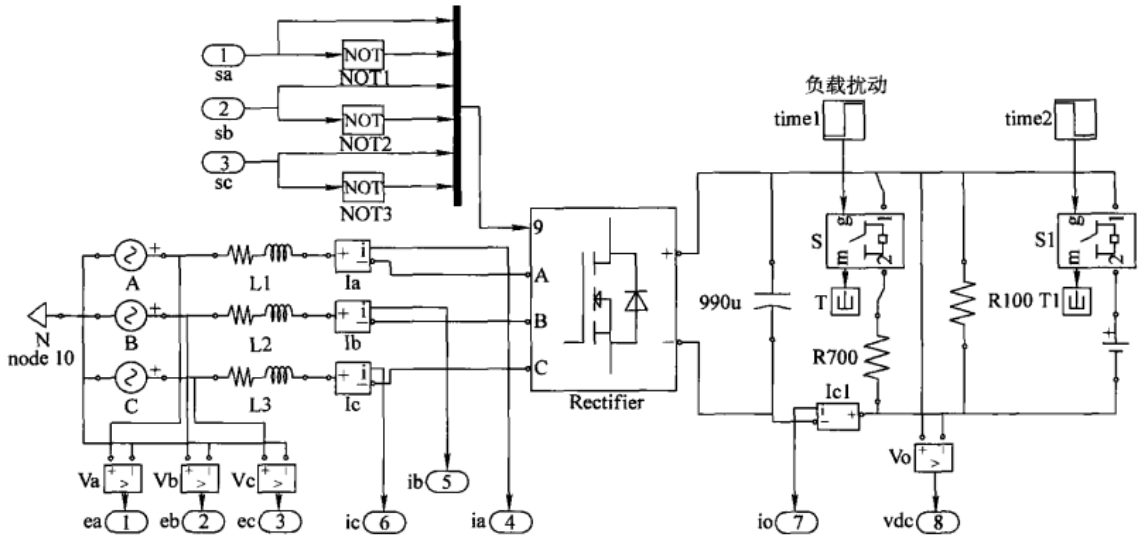


图 5-35　PWM 整流器的仿真模型

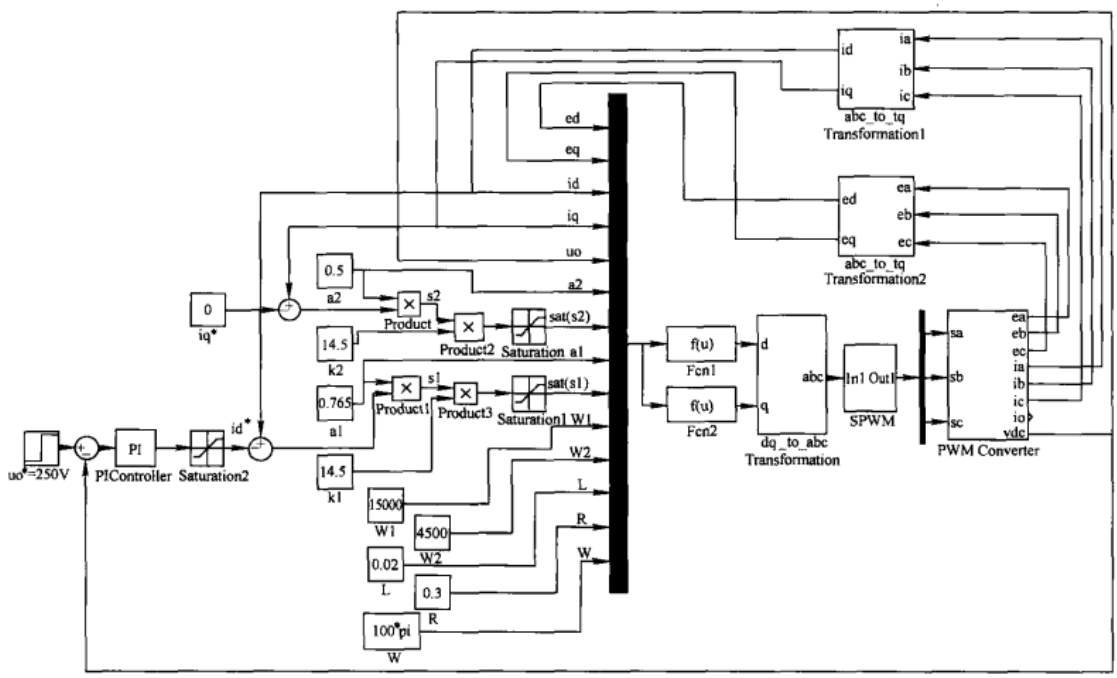


图 5-36　PWM 整流器滑模变结构控制系统仿真模型

出进行了饱和限幅处理。“dq_to_abc Transformation”为控制信号从同步旋转坐标系到三相静止坐标系的“等功率”坐标变换矩阵。“abc_to_dq Transformation1”和“abc_to_dq Transformation2”分别为三相电流和三相电压从三相静止坐标系到同步旋转坐标系的“等功率”坐标变换矩阵。“SPWM”为 PWM 调制模块，“PWM Converter”为被控对象（PWM 整流器）的仿真模型，是将图 5-35 所示的模型进一步封装得到。Fcn1 和 Fcn2 为式（5-24）所示的电流内环控制器，其值分别为

Fcn1:u[12]*(u[1]/u[12]-u[13]/u[12]*u[3]+u[14]*u[4]+u[10]*u[9]/

u[8])/u[5]

Fcn2:u[12]*(u[2]/u[12]-u[13]*u[4]/u[12]-u[14]*u[3]+u[11]*u[7]/u[6])/u[5]

对于 PWM 整流器，其参数 $R=0.3\Omega$，$L=0.02\text{H}$，$C=990\mu\text{F}$，$e_a=78\cos\omega t$，$e_b=78\cos(\omega t-120°)$，$e_c=78\cos(\omega t+120°)$，负载电阻 $R_L=100\Omega$。对于滑模变结构控制器，其参数 $\alpha_1=0.765$，$\alpha_2=0.5$，$W_1=15000$，$W_2=4500$，$k_1=k_2=14.5$。对于 PI 控制器，给定电压 $u_o^*=250\text{V}$，控制器参数 $K_{pi}=0.188$，$\tau_i=0.0687$，饱和限幅值 $U_{sat}=\pm10\text{V}$。需要注意的是，这里的 PI 参数是在式（5-30）、式（5-31）的理论计算基础上，进一步通过仿真实验得到的最佳参数。控制系统仿真实验结果如图 5-37 所示。

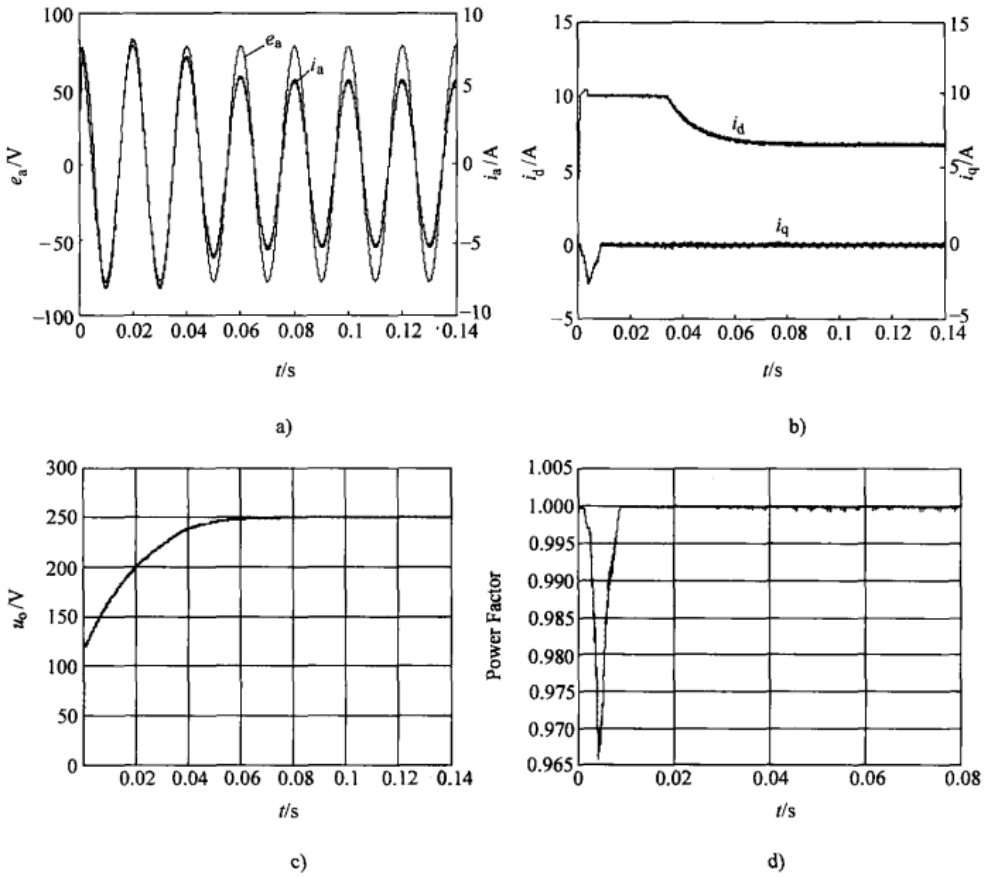


图 5-37 滑模变结构控制系统的仿真曲线

a）e_a，i_a 的波形 b）i_d，i_q 的波形

c）直流侧输出电压 u_o 的波形 d）功率因数波形

从上述仿真结果可以看出：①系统具有较快的动态响应速度，有功、无功电流和直流侧输出电压于 $t=0.08\text{s}$ 时达到稳态期望值；②从直流侧输出电压波形可以看出，稳态时输出直流电压无静差；③从功率因数波形可以看出，在 $t=0.01\text{s}$ 之前，由于此时存在电流波形畸变，导致功率因数较低（在 0.966 左右），达到稳态之后，功率因数为 0.9994，实现了单位功率因数控制。

为更好地验证 PWM 整流器控制系统性能，这里进一步进行负载扰动实验和给定电压突

变实验。在 $t=0.15$s 时，负载由 $R_L=100\Omega$ 突变为 $R_L=87.5\Omega$（即再并联一个 700Ω 的电阻）；在 $t=0.45$s 时，直流给定电压由 $u_o^*=250$V 突变为 $u_o^*=280$V，仿真实验结果如图 5-38 所示。从仿真实验结果可以看出，所设计的 PWM 整流器滑模变结构控制系统对于负载变化具有很好的抗扰性能，同时对于给定直流输出电压的变化表现出了很快的动态响应速度。

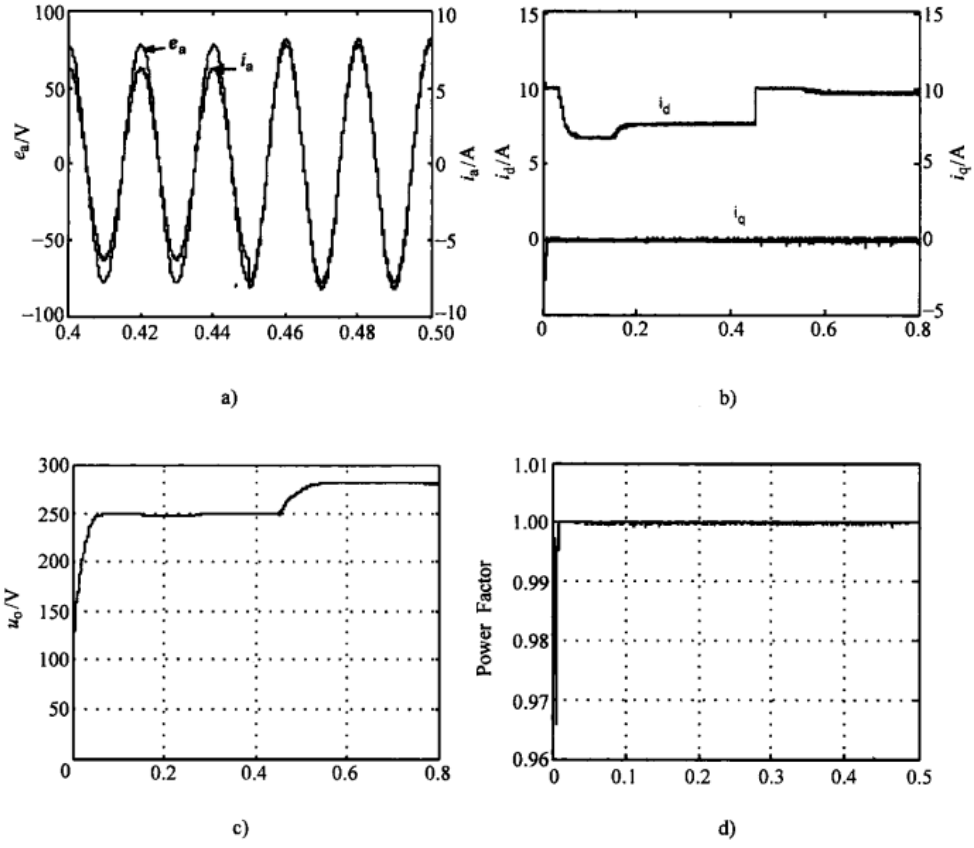


图 5-38　负载扰动及电压给定突变实验

a）e_a，i_a 的波形　b）i_d，i_q 的波形

c）直流侧输出电压 u_o 的波形　d）功率因数波形

（三）. 控制方案比较分析

对于上述实验结果，与传统的“双闭环 PID 控制方案”[99-101]相比较，可以得出以下几点结论：

1）滑模变结构控制系统的稳态/动态性能指标要优于传统 PID 控制。

由于滑模变结构控制的滑动模态可以自行设计，并且对于系统参数及扰动具有不变性，这就使得滑模变结构控制具有快速动态响应、物理实现简单、对参数变化及扰动不灵敏、鲁棒性强等优点。对于 PWM 整流器，这种强鲁棒性表现为对于负载扰动和电网电压扰动具有很强的抗扰性能。

2）与传统 PID 控制相比，滑模变结构控制器的参数整定更易于实现。

对于传统 PID 控制，按照“工程化参数整定方法”[6]计算得到的 PID 参数，在实际应用时仍需在较大范围内进一步调整，才可获得较好的控制系统性能指标，PID 控制器参数整定较为困难。随着智能控制理论的发展，模糊控制、神经网络控制、专家系统和遗传算

法[102-104]等理论被应用于 PID 控制器参数寻优中，但这又使得控制系统设计更加复杂，同时不利于工程实现。从本节的介绍可以看出，由于滑模变结构控制可实现系统大范围工作稳定，因此其参数整定较为容易。

建议读者可在 MATLAB/Simulink 环境下，分别进行两种控制方案的仿真实验，以此来体会两种控制方案在参数整定上的不同之处。

当然，滑模变结构控制也存在一些缺点，最突出的问题就是控制力的“抖振”现象。在滑模变结构控制的具体工程应用中，可以采用如前所述的一些“消抖”方法，使抖振幅度限定到工程允许范围内。然而，“抖振”现象只能在一定程度上抑制，而无法从根本上消除；消除了抖振，也就消除了滑模变结构控制所特有的强鲁棒性。

六、结论

本节通过对“三相电压型 PWM 整流器”滑模变结构控制系统的设计与仿真实验，可以得出以下几点结论：

1）本节以三相电压型 PWM 整流器为例，探讨了电力电子系统的建模、模型验证、控制器设计及控制系统仿真等问题；介绍了“SimPowerSystems 仿真工具箱”的基本情况及其使用方法，利用该工具箱可以较为方便地完成电力电子变换器控制系统仿真，加快电力电子系统研究/开发进度。

2）本节设计了基于滑模变结构控制的 PWM 整流器控制系统，较为系统地介绍了滑模变结构控制理论，总结归纳了滑模变结构控制系统的设计方法。它可概括为：首先，选择适当的滑模面参数，构成希望的滑动模态；其次，求取不连续控制 u^*，保证在切换平面 $s=0$ 上的每一点存在滑动模态，并使系统状态进入到滑动平面。同时，为将滑模变结构控制应用于实际系统中，需采取一定的“消抖”措施；本节所介绍的“边界层法”是一种简单而有效的消抖方法。

3）仿真实验结果表明，相比于传统的双闭环 PID 控制方法，本节所设计的滑模变结构控制系统具有较快的动态响应速度、较强的抗负载扰动能力和给定电压突变跟随性能，可实现单位功率因数控制。同时，该滑模变结构控制系统易于控制器参数整定，便于工程实现。

需要说明的是，本节所阐述的 PWM 整流器控制系统设计是在假定电网电压平衡的条件下完成的。然而，实际工况中三相电网电压在幅值和相位上往往是不平衡的。因而，研究“电网不平衡条件下”PWM 整流器的单位功率因数控制问题具有重要的实际意义，同时也具有很大的挑战性。

感兴趣的读者可参阅参考文献［105-107］，对此问题进行深入研究。

第六节　问题与探究——两轮电动车自平衡控制问题

一、问题提出

2001 年夏天的美国新罕布什尔州，一台外形奇特、能够自己保持平衡并随着驾乘者身体重心变化而作出相应运动的两轮电动车吸引了全世界关注的目光，人们称其为 Segway，如图 5-39 所示。Segway 之所以能够保持自身平衡的秘密在于其内置的 5 个精密陀螺仪，它

们实时地监控车身倾角状态并将其传给微处理器，后者通过分析接收到的状态信号计算出相应的指令，驱动电动机工作保持车身平衡。

Segway 的控制原理与一阶直线倒立摆是相似的，当车身（摆杆）前倾时，倾角传感器将这一状态传递给微控制器，微控制器通过内置算法计算出相应控制量驱动电动机运转使车身产生向前的一个加速度以保持自身的平衡状态，反之亦然。与此同时，由于两个车轮分别由相互独立的电动机控制，只要调整两轮转速使其产生转速差即可达到转弯目的，甚至可以完成 0 半径的原地旋转。

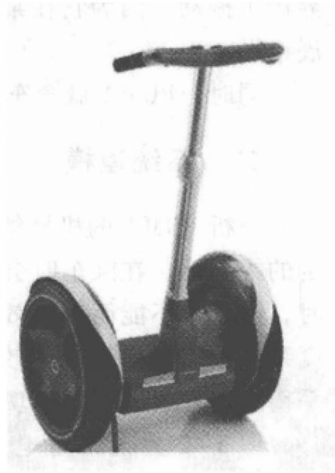


图 5-39　第一代 Segway 产品

相对于传统的城市交通工具，自平衡式两轮电动车由于它独特的机械结构和先进的控制理念有着很多突出的优点：体积小，甚至可以放入汽车的后备箱，这让它在拥挤的城市中如鱼得水；操作便捷，驾驶者能够随心所欲地控制车辆运行，车体仿佛是人体的一部分；清洁环保，电动车能源采用镍氢及锂电池，零排放、无污染，也使得它能进入机场、商场等密封的公共场所。因此，有人一度认为它将创造人类交通工具发展的新时代。

然而，一个新的事物的出现总会伴随着一些新的问题。面对这样一款另类的电动车，人们很难将其归入任何一类传统交通工具，因为在法律上它既不能在机动车道上行驶也不能在自行车道或人行道上行驶，一些城市不得不明令禁止市民驾乘其上路；另外，不菲的价格也使得它难于被大众接受。种种原因致使 Segway 在商业上并不十分成功。

面对窘境，公司也做出了相应改进以求让这颗新星能够得到更多的认可，他们相继推出了适于山地越野的 XT 系列和专为高尔夫球场设计的 GT 系列；与此同时在宣传上他们也做足功夫，Segway 在媒体和影视剧中频频亮相，相信看过《超市特警》一片的人都会被主人公那匹“良驹”深深吸引，如图 5-40 所示。



图 5-40　《超市特警》中的 Segway 剧照



图 5-41　Segway 与 GM 公司联手设计的 PUMA

2009 年初，Segway 联手 GM（通用汽车）公司推出了一款坐式 Segway 概念车——PUMA（见图 5-41），并计划于 2012 年上市。它在控制理念上继承了前几代 Segway 产品的思

想，而在外形及时速上更贴近普通机动车，并可两人同时乘坐。这一概念车一经提出就在业界产生振动，因为它在某种程度上弥补了原有 Segway 的缺陷，也将为 Segway 公司的产品拓展市场。

同时，PUMA 概念车的推出也为控制工程师们提出了许多挑战性的问题。

二、系统建模

分析 PUMA 的机械结构及人体的运动特点，可以得到以下结论：人的腰部及脊椎有一定的柔韧性，在乘车时会因惯性而出现前倾和后仰等情况，所以当驾乘者由站立改为坐姿时，其身体不能简单等效为一根直杆。本书用一段劲度系数较高的弹簧来等效人的腰椎，模拟人在乘坐自平衡式两轮电动车时出现的腰椎轻微弯曲变形。

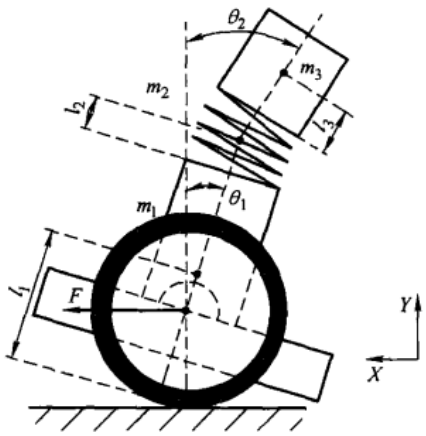


图 5-42 PUMA 系统等效图模型

图 5-42 为人乘坐电动车时的等效模型，车身包括：人体与车身接触紧密的腰椎以下部分，称为下杆，质量为 m_1，其质心到地面距离为 l_1；人的腰椎部分称为弹簧，质量为 m_2，其质心到下杆顶端距离为 l_2；脊椎以上部分称为上杆，质量为 m_3，其质心到弹簧顶端距离为 l_3。θ_1 为下杆倾角，θ_2 为上杆倾角，F 为系统外力，x_1 为系统位移。

为运用拉格朗日方程建模，首先求出车身各部分的速度，然后求取系统的动能和重力势能及广义力，最后代入方程求解。

下杆的动能为

$$T_1 = \frac{1}{2}m_1\left[(\dot{x}_1 - \dot{\theta}_1 l_1\cos\theta_1)^2 + (\dot{\theta}_1 l_1\sin\theta_1)^2 + \frac{1}{3}(2l_1\dot{\theta}_1)^2\right]$$

弹簧的动能为

$$T_2 = \frac{1}{2}m_2[\dot{x}_1 - \dot{\theta}_1(2l_1 + l_2)\cos\theta_1]^2 + \frac{1}{2}m_2[\dot{\theta}_1(2l_1 + l_2)\sin\theta_1]^2 + \frac{1}{4}m_2 l_2^2(\dot{\theta}_1^2 + \dot{\theta}_2^2)$$

上杆的动能为

$$T_3 = \frac{1}{2}m_3[\dot{x}_1 - \dot{\theta}_1(2l_1 + l_2)\cos\theta_1 - \dot{\theta}_2(l_2 + l_3)\cos\theta_2]^2 + \frac{1}{6}m_3(2l_3)^2\dot{\theta}_2^2$$

$$+ \frac{1}{2}m_3[\dot{\theta}_1(2l_1 + l_2)\sin\theta_1 + \dot{\theta}_2(l_2 + l_3)\sin\theta_2]^2$$

系统总动能为 $T = T_1 + T_2 + T_3$

系统的重力势能为

$$V = m_1 g l_1\cos\theta_1 + m_2 g(2l_1 + l_2)\cos\theta_1 + m_3 g[(2l_1 + l_2)\cos\theta_1 + (l_2 + l_3)\cos\theta_2]$$

选取广义坐标 θ_1、θ_2、x_1。在 θ_1 方向上系统不受广义力，x_1 方向上受到外力 F，θ_2 方向上受到弹簧弹力作用，设弹簧弯矩系数为 k，表达为 $k(\theta_2 - \theta_1)$，且系统会受到一定阻尼力作用 $f(\dot{\theta}_2 - \dot{\theta}_1)$。在三个方向上分别应用拉格朗日方程 $Q_i = \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i}$ 得到以下三个方程：

$$\ddot{\theta}_1\left[\frac{7}{3}m_1l_1^2+m_2\left(4l_1^2+4l_1l_2+\frac{3}{2}l_2^2\right)+m_3(2l_1+l_2)^2\right]$$
$$-\dot{\theta}_2^2m_3(2l_1+l_2)(l_2+l_3)\sin(\theta_2-\theta_1)+\ddot{\theta}_2m_3(2l_1+l_2)(l_2+l_3)\cos(\theta_2-\theta_1)$$
$$-(\ddot{x}_1\cos\theta_1+g\sin\theta_1)[m_1l_1+m_2(2l_1+l_2)+m_3(2l_1+l_2)]=0 \tag{5-32}$$

$$\ddot{\theta}_2\left[\frac{1}{2}m_2l_2^2+m_3(l_3+l_2)^2+\frac{4}{3}m_3l_3^2\right]+\ddot{\theta}_1m_3(2l_1+l_2)(l_2+l_3)\cos(\theta_2-\theta_1)$$
$$-\dot{\theta}_1^2m_3(2l_1+l_2)(l_2+l_3)\sin(\theta_2-\theta_1)-(\ddot{x}_1+g)m_3(l_2+l_3)(\cos\theta_2+\sin\theta_2)$$
$$=k(\theta_2-\theta_1)+f(\dot{\theta}_2-\dot{\theta}_1) \tag{5-33}$$

$$(m_1+m_2+m_3)\ddot{x}_1-[m_1l_1+(m_2+m_3)(2l_1+l_2)](\ddot{\theta}_1\cos\theta_1-\dot{\theta}_1^2\sin\theta_1)$$
$$-m_3(l_2+l_3)(\ddot{\theta}_2\cos\theta_2-\dot{\theta}_2^2\sin\theta_2)=F \tag{5-34}$$

以上式（5-32）、式（5-33）、式（5-34）即为系统的精确模型。

三、问题探究

（一）建模问题

式（5-32）、式（5-33）、式（5-34）给出的是系统模型的微分方程形式，为了便于人们应用已有的控制理论与方法来进行控制系统的设计，还需要我们作进一步的近似与线性化处理，建立起系统的传递函数模型，如图 5-43 所示。



图 5-43　系统传递函数模型

思考：图 5-43 中的 $G_1(s)$、$G_2(s)$ 与 $G_3(s)$ 具体表达式是什么？

（二）控制问题

如何实现图 5-41 所示“两轮电动车”的运动控制？

如何实现车体在自平衡条件下的直线运动与转弯运动？

建议：可以借鉴倒立摆的平衡控制方案或应用“人工智能控制”与“非线性控制”等理论方法进行探究。

（三）系统实现问题

搭建自平衡式两轮电动车实物系统时，我们将会遇到以下几个问题：

1）电动机的选取问题。考虑到整体系统的各项性能，选取何种类型、多大尺寸、多少功率的电动机才最为恰当呢？

2）倾角传感器的选择问题。为了实时精确地监测车身姿势状态，倾角传感器应当具有高精度与高频响等特性，那么该如何考虑具体的技术指标呢？

3）操控机构的选择问题。采用“方向盘”还是“操纵杆”来实现车体的直线与转弯运动？

以上问题，读者可以参考以下几个网站：

Segway 官方网站：http：//www. segway. com/；

平衡踏板车：http：//tlb. org/scooter. html；

MIT 的研制情况：http：//web. mit. edu/first/segway/；

PURDUE 的研制情况：http：//cobweb. ecn. purdue. edu/ ~ dsml/ece477/。

小　结

我们学习与掌握“控制系统数字仿真与 CAD”技术的目的在于有效地提高控制系统分析、设计与制造过程中的效率，以使最终的工作结果有所保障。

本章通过对几个实际问题的分析与仿真实验研究，总结了双闭环直流调速系统的控制原理，领会了建立过程控制对象（水箱液位控制问题）模型的意义，验证了有关参考文献介绍的控制算法（PID 调节器的鲁棒性设计方法），证明了所提出的控制策略（一阶倒立摆系统的双闭环模糊控制方案）的有效性，从中可以使我们进一步地认识“控制系统数字仿真与 CAD”技术的重要意义。

作为科学研究的具体形式之一，仿真实验的科学性与有效性需要不断地在实践中总结、领会与提高，只有对所研究问题全面、深入地理解与设计，才能使最终的仿真实验结果具有实际意义。在本章最后习题中给读者的几个问题都是从实际工作中抽象出来的，希望对大家的学习能够有所启迪和帮助。

习　题

5-1　如图 5-44 所示“旋转倒立摆系统”，试讨论如下问题：

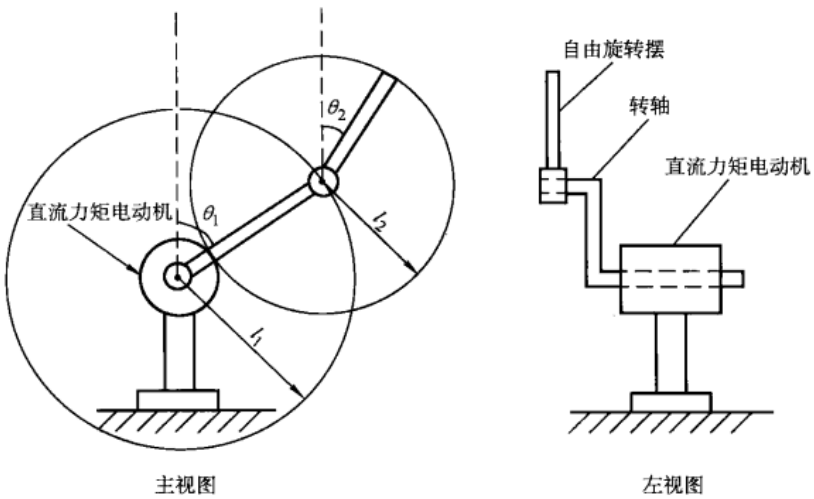


图 5-44　题 5-1 旋转倒立摆系统原理图

1）能否通过控制直流力矩电动机的输出转矩来实现“自由旋转摆”的垂直倒立（即 $\theta_1 = \theta_2 \approx 0$）？

2）能否实现“自由旋转摆”在平面的适当移动（即 $\theta_1 \approx c$（常数），$\theta_2 \approx 0$）？

3）试给出你的具体实现方案。

5-2　在如图 5-45 所示的某高精度齿轮测量机系统中，为保证主轴系统的回转精度，采用了“过盈量轴承”的装配技术，从而使主轴控制电动机在低速下的机械特性呈现图 5-45b 所示的情况（带有随机因素影响的非线性库仑摩擦特性）。若电动机选用低速大惯量永磁力矩电动机（基本参数：额定电压 36V，额定转速 30r/min、瞬时最大转矩 50N · m），回转角度检测选用分辨率为 0.25″（角度）的高精度圆光栅，试分析回答如下问题：

1）若系统要求控制电动机达到（1r/10h ~ 1r/min）的调速范围，试设计电控系统的主回路（即电动机

的驱动器）、数字式给定电路以及数字控制器。

2）若测量工艺要求伺服系统定位精度达到±1个脉冲（0.25″），定位步长为0.1°，试利用数字仿真技术设计该伺服系统，使得其动态性能具有“快速-无超调”的特点。

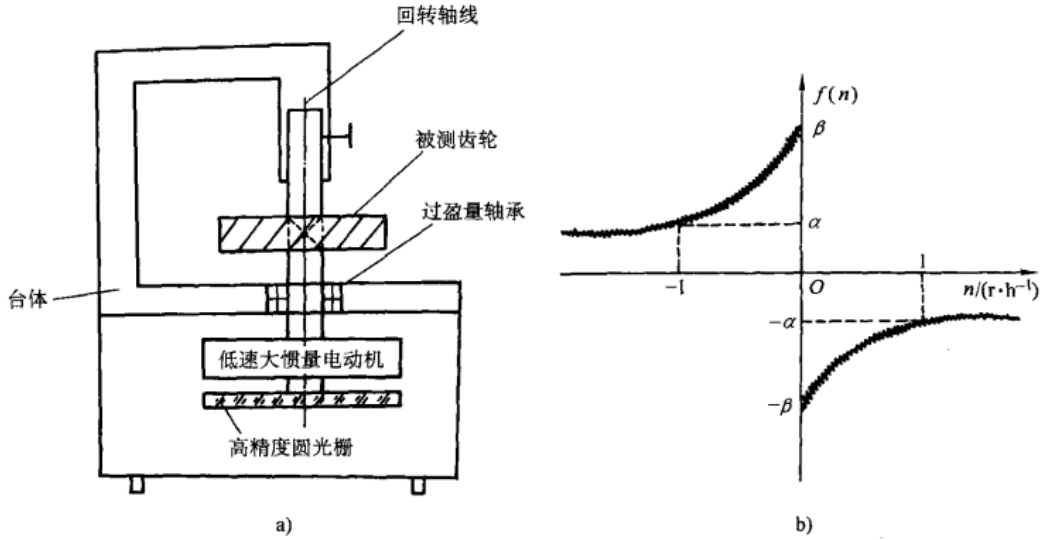


图5-45　题5-2高精度齿轮测量机主轴系统结构图

a）主轴系统　b）低速下的库仑摩擦特性

5-3　如今，在大型货场及码头的集装箱搬运中，广泛采用“龙门起重机”或“轮式搬运起重机”，我们将其“搬运—定位—安放”过程抽象为图5-46所示情况。若小车质量为m_1，控制作用为$F(t)$（可视为拖动力），重物质量为m_0，绳索长度为$L(t)$，试利用仿真实验的方法讨论如下问题：

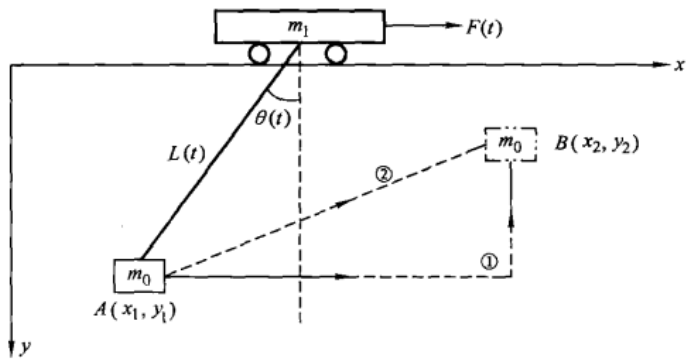


图5-46　题5-3龙门起重机重物定位过程示意图

1）把重物由A点提升至B点（或由B点放下至A点）的两种“负载位置伺服控制”方案哪种时间最短？（提示：方案1为“定摆长消摆位置控制方案”，方案2为“变摆长消摆位置控制方案”）

2）试给出你的具体实现方案。（提示：说明系统的实物制作原理与控制策略）

5-4　在大型立体化仓库中经常采用如图5-47所示的搬运机器人，由于H值较高，水平方向位置检测码盘又装在顶部，所以在定位过程中往往会产生“位置抖动”（用θ来描述）现象，从而影响定位的精度，降低了定位过程的效率。试针对这一问题研究如何建立最佳控制规律$u(t)$（运行速度），以使机器人从A点运行定位到B点所用的时间最短，同时又满足一定的定位精度要求。

（提示：本问题所讨论的问题可抽象为图5-48所示的物理模型——具有弹性立杆的移动小车问题）

5-5　设某32层公寓已有三部运载电梯（均由计算机控制），为提高其工作效率（能量消耗最少、候梯

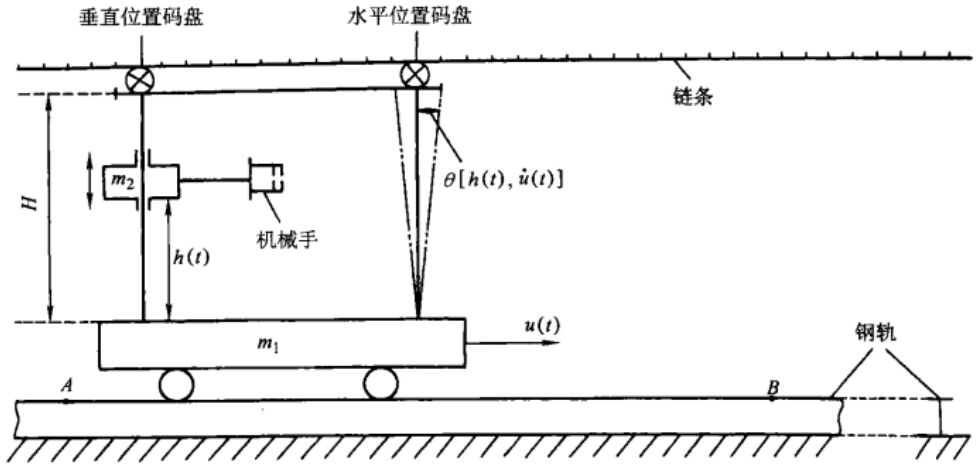


图 5-47 题 5-4 立体仓库搬运机器人结构图

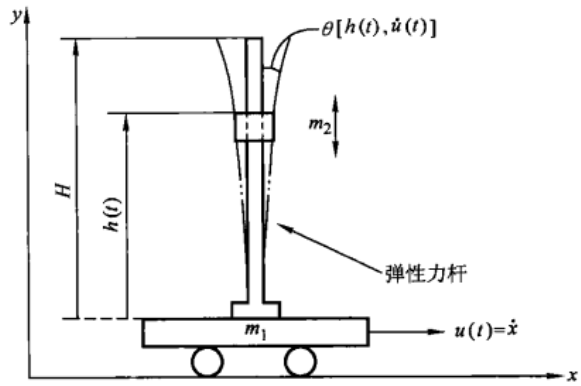


图 5-48 具有弹性立杆的移动小车问题示意图

时间最短），特增设管理与控制计算机一台，利用网络通信的方法将三部电梯连接起来。试设计管理计算机中的“电梯运行最佳调度”软件，以使该公寓电梯运行的效率达到最佳，并用数字仿真的方法证明之。

参 考 文 献

[1] 任兴权．控制系统计算机仿真［M］．北京：机械工业出版社，1988.

[2] 夏德钤．自动控制理论［M］．北京：机械工业出版社，1990.

[3] 曹克民．自动控制原理学习指导［M］．西安：西安地图出版社，1995.

[4] 徐士良．计算机常用算法［M］．北京：清华大学出版社，1995.

[5] 黄柯棣，等．系统仿真技术［M］．长沙：国防科技大学出版社，1998.

[6] 陈伯时．电力拖动自动控制系统运动控制系统［M］．3 版．北京：机械工业出版社，2004.

[7] 熊光楞．控制系统数字仿真［M］．北京：清华大学出版社，1982.

[8] 涂健．控制系统的数字仿真与计算机辅助设计［M］．武汉：华中工学院出版社，1985.

[9] 史忠科，卢京潮，等．自动控制原理常见题型解析及模拟题［M］．西安：西北工业大学出版社，1998.

[10] 韩九强．MATLAB 高级语言及其在控制系统中的应用［M］．西安：西安交通大学出版社，1997.

[11] 施阳，李俊，等．MATLAB 语言工具箱——TOOLBOX 实用指南［M］．西安：西北工业大学出版社，1998.

[12] 张志涌，刘瑞桢，杨祖樱，等．掌握和精通 MATLAB［M］．北京：北京航空航天大学出版社，1998.

[13] 张培强．MATLAB 语言——演算纸式的科学工程计算语言［M］．合肥：中国科学技术大学出版社，1995.

[14] 薛定宇．控制系统计算机辅助设计——MATLAB 语言及应用［M］．北京：清华大学出版社，1996.

[15] 楼顺天，于卫，等．基于 MATLAB 的系统分析与设计——控制系统［M］．西安：西安电子科技大学出版社，1998.

[16] 楼顺天，李博菡．基于 MATLAB 的系统分析与设计——信号处理［M］．西安：西安电子科技大学出版社，1998.

[17] 施阳，等．MATLAB 语言精要及动态仿真工具 SIMULINK［M］．西安：西安工业大学出版社，1997.

[18] 楼顺天，于卫，闫华梁，等．MATLAB 程序设计语言［M］．西安：西安电子科技大学出版社，1997.

[19] 陈怀琛，黄道君，等．控制系统 CAD 及 MATLAB 语言［M］．北京：电子工业大学出版社，1996.

[20] S 巴斯．计算机算法：设计和分析引论［M］．朱洪，等译．上海：复旦大学出版社，1985.

[21] 李庆杨，等．数值分析［M］．武汉：华中理工大学出版社，1986.

[22] WH 普雷斯，等．数值方法大全——科学计算的艺术［M］．王璞，等译．兰州：兰州大学出版社，1991.

[23] KE 阿特金森．数值分析引论［M］．匡蛟勋，等译．上海：上海科技大学出版社，1986.

[24] 叶庆凯．控制系统计算机辅助设计［M］．北京：北京大学出版社，1990.

[25] 翁思义．控制系统计算机仿真与辅助设计［M］．西安：西安交通大学出版社，1987.

[26] 韩慧君．系统仿真［M］．北京：国防工业出版社，1985.

[27] 吴重光．控制系统计算机辅助设计［M］．北京：机械工业出版社，1986.

[28] 吴智铭．控制系统计算机辅助设计［M］．北京：电子工业出版社，1986.

[29] Duance anselman，Bruce Littelfield. 精通 MATLAB 综合辅导与指南［M］．李人厚，张平安，译．西

安：西安交通大学出版社，1998.
[30] KJ 奥斯特隆姆，B 威顿马克．计算机控制系统理论与设计［M］．北京：科学出版社，1987.
[31] 周斌，等．自动控制系统实验技术［M］．北京：机械工业出版社，1986.
[32] Cs Banyasi. An adaptive PID regulator dedicated for microprocessor lased Compart Contollers［J］. The 7th IFAC Symposium of Identification and system paramter Estimation, Preprints.
[33] 李士勇．模糊控制·神经控制和智能控制论［M］．哈尔滨：哈尔滨工业大学出版社，1996.
[34] 冯纯伯，史维．自适应控制［M］．北京：电子工业出版社，1986.
[35] 袁震东．自适应控制理论及其应用［M］．上海：华东师范大学出版社，1988.
[36] 金先级．机电系统的计算机仿真［M］．北京：清华大学出版社，1991.
[37] 王正中．系统仿真技术［M］．北京：科学出版社，1986.
[38] 江裕钊．数学模型与计算机模拟［M］．西安：电子科技大学出版社，1989.
[39] 汪端芳．连续系统仿真及应用技术［M］．重庆：重庆大学出版社，1986.
[40] 薛嘉庆．最优化原理与方法［M］．北京：冶金工业出版社，1986.
[41] 陈佳实．微机控制与微机自适应控制［M］．北京：电子工业出版社，1987.
[42] 张晓华．伺服系统调节器参数优化设计的一种直接方法［J］．机械工业自动化，1989，(4)．
[43] Zhang Xiaohua, Zhou Xiao. Research on Predictive Fuzzy Control of Pipeline Robot［J］. Journal of HIT, 1998, 5 (2)．
[44] 张晓华，付庄．基于视觉传感器的管内作业机器人位置检测［J］．机器人，1997，19 (5)．
[45] 张晓华，王卓军．立体库搬运机械手逻辑控制系统设计［J］．电气自动化，1997，19 (3)．
[46] Katsuhiko Ogata. 现代控制工程［M］.3 版．北京：电子工业出版社，2000.
[47] Richard C Dorf. 现代控制系统［M］.8 版．北京：高等教育出版社，2001.
[48] Mohand Mokhtari. MATLAB 与 SIMULINK 工程应用［M］．北京：电子工业出版社，2002.
[49] Gene F Franklin. 动态系统的反馈控制［M］.4 版．北京：电子工业出版社，2004.
[50] 薛定宇．反馈控制系统设计与分析——MATLAB 语言应用［M］．北京：清华大学出版社，2000.
[51] 薛定宇．基于 MATLAB/SIMULINK 的系统仿真技术与应用［M］．北京：清华大学出版社，2002.
[52] 迪安 K 费雷德里克．反馈控制问题—使用 MATLAB 及其控制系统工具箱［M］．西安：西安交通大学出版社，2001.
[53] 末松良一．机械控制入门［M］．北京：科学出版社，2000.
[54] 细江繁幸．系统与控制［M］．北京：科学出版社，2001.
[55] 王子才．应用最优控制［M］．哈尔滨：哈尔滨工业大学出版社，1989.
[56] 蔡增威．一阶直线倒立摆运动控制技术的研究［D］．哈尔滨：哈尔滨工业大学．2004.
[57] 熊永波．吊车防摆实物仿真技术研究［D］．哈尔滨：哈尔滨工业大学，2003.
[58] 哈尔滨工业大学理论力学教研室．理论力学：下册［M］.3 版．北京：高等教育出版社，2000.
[59] 张晓华．系统建模与仿真［M］．北京：清华大学出版社，2006.
[60] 郑建容．ADAMS-虚拟样机技术入门与提高［M］．北京：机械工业出版社，2002.
[61] 王国强，张进平，马若丁．虚拟样机技术及其在 ADAMS 上的实践［M］．西安：西北工业大学出版社，2002..
[62] 李虎林，易湘斌．产品开发中的多软件联合仿真技术［J］．机械设计与制造，2008 (4)：55-56.
[63] 王晓东，毕开波，周须峰．基于 ADAMS 与 Simulink 的协同仿真技术及应用［J］．计算机仿真，2007，24 (4)：271-274.
[64] 张崇巍，张兴．PWM 整流器及其控制［M］．北京：机械工业出版社，2003.
[65] Tzann-shin Lee. Lagrangian Modeling and Passivity-Based Control of Three Phase AC/DC Voltage-Source Converters［J］. IEEE Transactions on Industrial Electronics, 2004, 51 (4)：892-902.

[66] Ortega R, Loria A, Nicklasson P J. Passivity-based Control of Euler-Lagrange Systems: Mechanical, Electrical and Electromechanical Applications [M]. London: Springer-Verlag, 1998.

[67] Ortega R, Schaft A, Maschke B, et al. Interconnection and Damping Assignment Passivity-based Control of Port-controlled Hamiltonian Systems [J]. Automatica, 2002, 38 (4): 585-596.

[68] 鞠儒生，陈宝贤，陈燕. 一种新型 PWM 整流器 [J]. 电工技术学报，2002，17 (6)：48-52.

[69] 徐德鸿. 电力电子系统建模及控制 [M]. 北京：机械工业出版社，2006.

[70] 蔡宣三. PWM 开关稳压电源的瞬态分析与综合 (一) [J]. 电力电子，2006 (2)：47-50.

[71] 蔡宣三. PWM 开关稳压电源的瞬态分析与综合 (二) [J]. 电力电子，2006 (3)：56-63.

[72] 蔡宣三. PWM 开关稳压电源的瞬态分析与综合 (三) [J]. 电力电子，2006 (4)：54-59.

[73] 冯慈璋，马西奎. 工程电磁场导论 [M]. 北京：高等教育出版社，2000.

[74] 吕峰，丁予展，高澜庆. 磁悬浮轴承的原理和应用 [J]. 机械，1994，21 (6)：40-43.

[75] 朱焜秋，等. 永磁偏置径向一轴向磁悬浮轴承工作原理和参数设计 [J]. 中国电机工程学报，2002，22 (9)：54-58.

[76] 曹建荣，虞烈，谢友柏. 主动磁悬浮轴承的解耦控制 [J]. 西安交通大学学报，1999，33 (12)：44-46.

[77] 邱杰，原渭兰. 数字计算机仿真中消除代数环问题的研究 [J]. 计算机仿真，2003，20 (7)：33-35.

[78] 姚俊，马松辉. SIMULINK 建模与仿真 [M]. 西安：西安电子科技大学出版社，2002.

[79] 郭云芳，等. 计算机仿真技术 [M]. 北京：北京航空航天大学出版社，1991.

[80] 耿华，杨耕. 控制系统仿真的代数环问题及其消除方法 [J]. 电机与控制学报，2006，10 (16)：632-635.

[81] Wujianqiang Lihaoyu. An IGBT Model Base on Study of Nonlinear Capacitances [J]. Journal of Harbin Institute of Technology, 1999, 6 (2): 20-24.

[82] Allen R Hefner. Modeling Buffer Layer IGBT' s for Circuit Simulation [J]. IEEE Trans On P E, 1995, 10 (2): 111-123.

[83] Allen R Hefner. Analytical Modeling of Device-Circuit Interactions for the Power Insulated Gate Bipolar Transistor (IGBT) [J]. IEEE Trans on I A, 1990, 26 (6): 995-1005.

[84] Chang SU Mitter, Allen R Hefner, Dan Y Chen, et al. Insulated Gate Bipolar Transistor (IGBT) Modeling Using IG-Spice [J]. IEEE Trans on I A, 1993, 30 (1): 24-33.

[85] Allen R Hefner. A Dynamic Electro-Thermal Model for the IGBT [J]. IEEE Trans on I A, 1994, 30 (2): 294-405.

[86] Allen R Hefner, David L Blackburn. Simulating the Dynamic Electrothermal Behavior of Power Electronic Circuits and Systems [J]. IEEE Trans on P E, 1993, 8 (4): 376-385.

[87] Hefnertt R, Blackburnt D L, Gallowaytt K F. The Effect Of Neutrons On The Characteristics Of The Insulaed Gate Bipolar Transistor (IGBT) [J]. IEEE Trans on N S, 1986, 33 (6): 1428-1434.

[88] 向博，高丙团，张晓华，等. 非连续系统的 Simulink 仿真方法研究 [J]. 系统仿真学报. 2006，18 (7)：1750-1754、1762.

[89] 华克强，高淑玲，朱齐丹. 吊车防摆技术的研究 [J]. 控制理论与应用，1992 (6)：631-637.

[90] 高为炳. 变结构控制基础 [M]. 北京：科学出版社，1989.

[91] 向博. 基于滑模变结构的吊车防摆技术研究 [D]. 哈尔滨：哈尔滨工业大学，2005.

[92] Bernhard Sprenger, Ladislav Kucera, Safer Mourad. Balancing of an Inverted Pendulum with a SCARA Robot [J]. IEEE/ASME Transactions on mechatronics, 1998, 3 (2): 91-97.

[93] William G Hurley. Electromagnetic Design of a Magnetic Suspension System [J]. IEEE Transactions on

education, 1997, 40 (2): 124-130.

[94] Jose Maria Giron-Sierra. A Simple Device and a Project for the Nonlinear Control Systems Laboratory [J]. IEEE Transactions on education, 2001, 44 (2): 144-150.

[95] Ka C Cheok, Nan K Loh. A Ball-Balancing Demonstration of Optimal and Disturbance Accommodating Control [J]. IEEE Control systems magazine, 1987.

[96] Jan Jantzen. Analysis of a Pendulum Problem [J]. Tech report no 98-E 863 (cartball), 1998 (8).

[97] 赵葵银. PWM 整流器的模糊滑模变结构控制 [J]. 电工技术学报, 2006, 21 (7): 49-53.

[98] Fernando J Silva. Sliding-Mode Control of Boost-Type Unity-Power-Factor PWM Rectifiers [J]. IEEE Transactions on Industrial Electronics, 1999, 46 (3): 594-603.

[99] 杨德刚, 赵良炳, 刘润生. 三相高功率因数整流器的建模及闭环控制 [J]. 电力电子技术, 1999, 33 (5): 49-51.

[100] 朱永亮, 马惠, 张宗濂. 三相高功率因数 PWM 整流器双闭环控制系统设计 [J]. 电力自动化设备, 2006, 26 (11): 87-91.

[101] 熊健, 张凯, 陈坚. PWM 整流器的控制器工程化设计方法 [J]. 电工电能新技术, 2002, 21 (3): 44-48.

[102] 高俊山, 牟晓光, 杨嘉祥. 一种基于遗传算法和神经网络的 PID 控制 [J]. 电机与控制学报, 2004, 8 (2): 108-112.

[103] 王伟, 张晶涛, 柴天佑. PID 参数先进整定方法综述 [J]. 自动化学报, 2000, 26 (3): 347-355.

[104] 韩华, 罗安, 杨勇. 一种基于遗传算法的非线性 PID 控制器 [J]. 控制与决策, 2005, 20 (4): 448-450, 454.

[105] 谭广军, 郭源博, 张晓华. 电网不平衡条件下三相电压型 PWM 整流器非线性控制研究 [J]. 电力电子, 2008 (3): 19-25.

[106] Song H, Nam K. Dual Current Control Scheme for PWM Converter under Unbalanced Input Voltage Conditions [J]. IEEE Transactions on Industrial Electronics, 1999, 46 (5): 953-959.

[107] Yin Bo, Oruganti Ramesh, Panda Sanjib Kumar, et al. An output-power-control Strategy for a Three-phase PWM Rectifier under Unbalanced Supply Conditions [J]. IEEE Transactions on Industrial Electronics, 2008, 55 (5): 2140-2151.